Waseda University

Doctoral Thesis

A study on speaker clustering considering inner/inter segment structure

発話内・発話間構造を考慮した話者クラスタリング手法の研究

February 2017

Naohiro TAWARA
俵　直弘

Waseda University

Doctoral Thesis

A study on speaker clustering considering inner/inter segment structure

発話内・発話間構造を考慮した話者クラスタリング手法の研究

Graduate School of Fundamental Science and Engineering
Department of Computer Science and Engineering
Research on Perceptual Computing

February 2017

Naohiro TAWARA
俵　直弘

Abstract

Recent progress in data archiving techniques has increased the demand for finding desirable data among the archives by using their attributes as queries. In this situation, we are faced with the problem of how to automatically provide these attributes with archived speech data, which are growing from day to day. Clustering is one of the key techniques for analyzing such a large amount of data. This thesis focuses on the clustering problem of multimedia data derived from the real world. One of the properties of multimedia data is that each data set is composed of a set of sequential observations. Acoustic signals and videos, for example, are composed of a set of acoustic frame-wise observations and a set of pixel-wise observations, respectively.

To cluster these segment-wise data, the following problems must be addressed:

- An individual frame-wise observation does not have enough information to distinguish a segment from other segments.
- The length of each sentence (i.e., the number of frame-wise observations in each sentence) differs from segment to segment.
- Each segment has a large variation caused by various factors such as noise.

In order to deal with the first problem, the stochastic properties of each segment must be taken into account. The second and third problems require the model to be robust against overfitting, allowing the model to be sufficiently flexible represent variations. In spite of their importance, these problems have received less attention for the clustering of segment-wise data. In this thesis, therefore, we explore segment clustering methods from the view points of these problems.

This thesis introduce two different segment-based clustering algorithms. The first approach is based on a segment generative model

that explicitly estimates the structure of mixture distributions in a fully Bayesian manner. In this approach, we first show that the optimal assignment of segments to clusters could be obtained by an estimated Gaussian mixture model from segment-wise data. We show that this model could be robustly estimated by introducing a Markov-chain Monte Carlo (MCMC)-based approach. We also show that the number of clusters could be estimated by introducing a nonparametric approach to this model. We then extend this segment-oriented model to a more complex model in which each component of a mixture model is also represented by a mixture model. The derived model is called the mixture of Gaussian mixtures models (MoGMMs), and we show that the MoGMMs is effectively estimated by an MCMC-based method called nested Gibbs sampling. We demonstrate its effectiveness in speaker clustering experiments in noisy situations

The second approach is based on the maximum a posteriori approach. In this model, a lower computational cost is required compared with the first approach because only the mean vectors of each cluster's GMM are utilized. We experimentally show that this approach often fails to estimate when each sample has a large perturbation caused by noise or statistical mismatch. We show that this problem can be solved by introducing a spectral clustering method. We apply the proposed approach to the speaker clustering problem in various noisy situations and show that it yields significant gains from conventional agglomerative and i-vector-based k-means clustering.

Contents

List of Tables

List of Figures

List of Abbreviations

ACP	Average cluster purity
ASP	Average speaker purity
BIC	Bayesian information criteria
CSJ	Corpus of spontaneous Japanese
CLR	Cross likelihood ratio
DER	Diarization error rate
EM	Expectation-Maximization
GMM	Gaussian mixtures model
HMM	Hidden Markov model
IV	i-vector
JNAS	Japanese Newspaper Article Sentences
JEIDA	Japan Electronic Industry Development Association
KL	Kullback-Leibler
LDA	Linear Discriminant Analysis
LML	Logarithmic marginalized likelihood
MoGMMs	mixture of Gaussian mixture models
MAP	Maximum a Posteriori
MCMC	Markov chain Monte Carlo
MFCC	Mel Frequency Cepstrum Coefficients
ML	Maximum Likelihood
NIST	National Institute of Standards and Technology
SC	Spectral clustering
SNR	Signal-to-noise ratio
SO-DPMM	Segment-oriented Dirichlet process mixture model
SRE	Speaker Recognition Evaluation
VB	variational Bayesian
UBM	Universal Background Model
WCCN	Within class covariant normalization

Chapter 1

Introduction

1.1 Background

With the considerable developments in data archiving techniques and computer resources, we can easily access a huge amount of data. If we need videos and music, for example, we can obtain a nearly unlimited amount of data from video-hosting websites. Most of these data, however, are not labeled with meta-information such as ”what appeared in the picture” and ”Who spoke in the video.” This situation increases the demand to find desirable data among the archives by using their attributes as queries. In this situation, we are faced with the problem of automatically providing these attributes with archived data. Clustering is an essential technique to solve this issue. By clustering archived data, we can estimate which data have the same attributes and which have the different ones.

Moreover, these readily accessible big data are also playing an important role in today’s development of artificial intelligence. With the help of such big data, machine learning technologies are developing at a surprising speed, achieving drastic improvements in various fields. While we are in this big data era, we are still struggling to make use of them. One reason for this is that most easily accessible data are unlabeled, although many traditional supervised machine learning systems require explicitly labeled examples. To make use of these unlabeled data, we are spending a large amounts of money and time to provide their appropriate labels. Clustering plays an important role in promoting the use of these unlabeled data. By clustering unlabeled data with previously labeled ones, for example, we can efficiently estimate the labels of these unlabeled data. And information in which the data are

the same or different will help us to understand unlabeled data. In this thesis, therefore, we explore the power of the clustering technique.

1.2 Problem definition

1.2.1 Segment clustering via a segment-oriented generative modeling

In this thesis, we focus on the clustering of multimedia data. These real-world data often comprise a set of component observations such as videos made of frame pictures and speech comprising frame-wise observations. One of the most important characteristics of these data is that they have a hierarchical structure, as illustrated in Fig 1.1. For example, in speech data obtained from a multiparty conversation, higher-level observations correspond to each speaker's utterances, where their variation is caused by the differences in the speakers. Lower-level observations, on the other hand, correspond to frame-wise observations comprising each segment, where their variation is caused by the differences in the contents of the speech. One ultimate goal of this thesis is to establish a clustering technology that can robustly cluster segments with interesting attributions that are of interest (e.g., the speakers of speech) independently of the attributions of disinterest (e.g., the contents of speech).

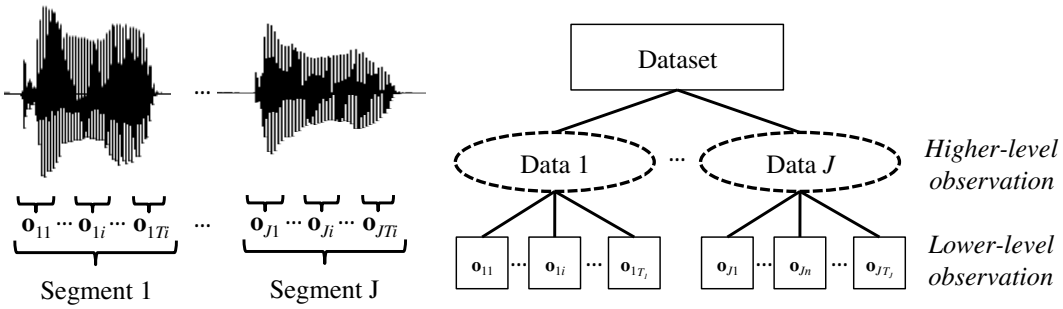


Figure 1.1: Hierarchical structure of multilevel data analysis. Segment-wise (higher-level) observations are composed of a set of frame-wise (lower-level) observations. The left figure illustrates the hierarchical structure in speech data composed of frame-wise observations (e.g. mel-frequency cepstral coefficients).

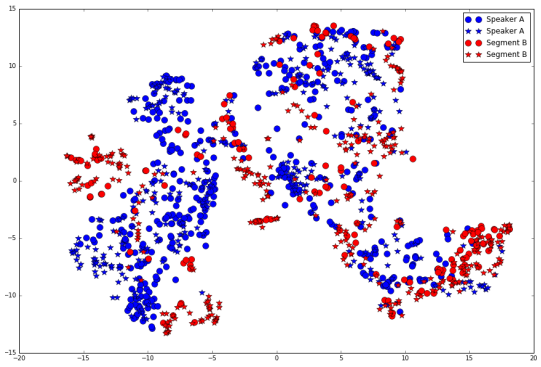


Figure 1.2: Frame-wise observations of an acoustic segment. The difference of shapes and colors correspond to the differences in the segments and speakers, respectively.

Fig 1.2 shows an example of segment-wise observations derived from acoustic segments spoken by two speakers. The plot corresponds to the individual frame-wise features projected onto a two-dimensional space using t-distributed stochastic neighbor embedding (t-SNE) [2], and different colors correspond to different speakers. From this figure, we can see that each segment has a unique distribution that generates each speaker's frame-wise observations. The simplest approach to representing each segment is to utilize a centroid of the segment as its landmark. This approach, however, is apparently insufficient because the inner variance of the segment is never considered. Instead, we need to derive a suitable mathematical representation of a segment for extracting each speaker's characteristics independently of the contents of their segment. An effective approach for representing segments is modeling them with stochastic distributions. Thus, we assume that each higher-level observation follows a unique distribution, which represents each speaker's characteristics.

Let $\mathbf{O}_u$ be the u-th segment comprising the T_u frame-wise observations $\{\mathbf{o}_{ut}\}_{t=1}^{T_u}$. Assuming that segments are independently generated from one of the C clusters with a frequency of h_i, the following mixture model is defined over a set of segments $\mathcal{O}=\{\boldsymbol{O}_u\}_{u=1}^{U}$;

$$p(\mathcal{O}) = \prod_{u=1}^{U} \sum_{i=1}^{C} h_i p(\mathbf{O}_u|\mathbf{\Theta}_i), \tag{1.1}$$

where $p(\mathbf{O}_u|\mathbf{\Theta}_i)$, h_i, and C denote a stochastic model of the i-th cluster, its

weight, and the number of clusters, respectively. One of the important aspects of this generative model is that a unit of observation is not a frame but a segment. This means that all of the frame-wise observations in a segment are simultaneously aligned to the same distribution. Frame-wise observations are then independently drawn from $p(\mathbf{O}_u|\mathbf{\Theta}_i)$. We call this hierarchical structure a "segment-oriented generative model."

By introducing this segment-oriented generative model, the optimal assignment of segments to clusters can be obtained by evaluating the posterior probability of assigning each segment to each cluster's distribution. Thus, the clustering problem is reduced to the problem of estimating this segment-oriented generative model.

1.3 Goal

One goal of the work presented in this thesis is to develop a segment generative approach that robustly works in a situation where segments are corrupted by nuisance information such as background noise. In order to achieve this purpose, we try to leverage the prior knowledge of a model structure using Bayesian estimation. In our approach, the structure of the segment is obtained by adapting the prior knowledge of its structure. By teaching this fundamental structure in model estimation, our approach can robustly estimate the case where each observation is seriously corrupted.

1.4 Overview

Here, we summarize the overview of our contributions in this thesis. Chapter 2 defines segment generative model and provides an overview of how to estimate this model. We show the conventional hierarchical agglomerative clustering (HAC)-based approach have several intrinsic drawbacks and introduce an alternative approach based on mixture modeling. Chapters 3 and 4 provide a detailed discussion about segment-oriented generative models. In these chapters, we develop a

fully Bayesian segment-generative model that has the following structure.

- **Single Gaussian mixture model:** In Chapter 3, we introduce a single Gaussian distribution to model each segment. Using this model, we explore the potential of sampling-based model estimation and show that this approach outperforms a conventional hierarchical agglomerative clustering (HAC)-based approach. We also show that the optimal number of speakers is also estimated by introducing a nonparametric prior called a Dirichlet process.

- **Mixture-of-mixtures modeling:** In Chapter 4, we introduce a Gaussian mixture model (GMM) to model each segment. We show that this model has a hierarchical structure that represents the segment-level data structure based on elemental mixture distributions and the higher-level (coarse) data structure based on mixture-of-mixtures distributions. We explore a suitable estimate for this model, and propose a novel sampling-based method called nested Gibbs sampling. We apply this method to a speaker clustering problem and conduct experiments under various conditions. The results demonstrate that the proposed method outperforms conventional sampling-based, variational Bayesian, and hierarchical agglomerative methods.

Permitting a heavy computational cost, this approach can provide a fine estimate of the segment-oriented generative model. However, this is not scalable to large data because the number of computation exponentially increases with the amount of data. In order to solve this problem, Chapter 5 provides an alternative approach inspired by an i-vector-based approach that is widely known in the speaker recognition community. In this approach, each segment is modeled by a GMM, as in the generative-based approach. However, only the mean vector is utilized to represent each segment, while the whole distribution is used in the aforementioned approach. The obtained mean vector is embedded into a more discriminative space, and the obtained vector is called i-vector. This approach provides nearly perfect performance

when each segment is the same as the prior information. The performance of this approach, however, considerably deteriorates in noisy situations because the statistical mismatch between the prior knowledge and the testing data causes serous distortion in the estimated i-vectors. In order to solve this problem, Chapter 5 introduces a spectral-based approach that can remove the noise-derived component from the similarity matrix between i-vectors. The proposed method is evaluated for the speaker clustering problem under very noisy situations and shows an overwhelming improvement compared with the conventional method. Finally, Chapter 6 reviews this thesis and discusses the future work based on the proposed methods.

Chapter 2

Formulations of segment-oriented clustering

This chapter begins by presenting a detailed definition of a segment-oriented generative model in Section 2.1. In Section 2.2, we overview conventional hierarchical agglomerative approaches that directly obtain each cluster's generative model. Then, in Section 2.3, we provide the general solution for estimating the model structure of a segment-oriented generative model in a Bayesian manner.

2.1 Segment-oriented generative model via mixture modeling

This section formalizes a segment-oriented generative model and explains how to estimate the structure of this model. Let $\mathbf{O}_u$ be the u-th segment comprising the T_u frame-wise observations $\{\mathbf{o}_{ut}\}_{t=1}^{T_u}$. Assuming that these segments are independently generated from one of the C clusters with a frequency of h_i, the following mixture model is defined over a set of segments $\mathcal{O}=\{\boldsymbol{O}_u\}_{u=1}^{U}$:

$$p(\mathcal{O}) = \prod_{u=1}^{U} \sum_{i=1}^{S} h_i p(\mathbf{O}_u|\boldsymbol{\Theta}_i), \tag{2.1}$$

where $p(\mathbf{O}_u|\boldsymbol{\Theta}_i)$, h_i and S denote a stochastic model of the i-th cluster, the corresponding weight, and the number of clusters, respectively. The frame-wise observations are then independently drawn from $p(\mathbf{O}_u|\boldsymbol{\Theta}_i)$.

In the next section, we provide relatively heuristic approaches that directly estimate model parameters and assign a segment a to cluster in an agglomerative manner.

2.2 Segment-oriented generative model via hierarchical agglomerative approaches

This section overviews model estimation approaches based on an agglomerative merging strategy. In this approach, each segment is modeled as a stochastic distribution, as in the segment generative approach. The optimal parameters of generative models are obtained by iteratively merging the most similar pair of distributions until some criterion is met. Fig 2.1 depicts the merging procedure in the HAC algorithm, in which clusters C_b and C_b are merged into a new cluster $C_{a \bigcup b}$. After merging these clusters, the parameters of this new cluster is estimated in maximum likelihood method. This approach is called HAC, has been broadly applied in various fields, and achieves state-of-the-art performance.

The earliest study of the HAC-based approach was carried out by Siegler et al [3]. In their work, each segment is modeled as a Gaussian distribution by maximum likelihood criteria. The similarity between segments is measured by the following generalized Kullback-Leibler (KL) (a.k.a. KL2) divergence:

$$\begin{aligned} KL2(i,j) &= KL(i,j) + KL(j,i) \\ &= \frac{\sigma_i^2}{\sigma_j^2} + \frac{\sigma_j^2}{\sigma_i^2} + (\mu_i - \mu_j)^2 \left(\frac{1}{\sigma_i^2} + \frac{1}{\sigma_j^2} \right), \end{aligned} \tag{2.2}$$

where μ_i, μ_j,σ_i^2, and σ_j^2 denote means and variances of the i-th and j-th clusters. $KL(i,j)$ denotes the KL divergence of the i-th cluster from the j-th cluster. Clusters with a minimum $KL2$ distance are iteratively merged while the minimum distance is larger than a predefined threshold.

Chen et al, on the other hand, proposed an another approach focusing on the estimation of the optimal number of clusters [4]. In this method, the similarity between clusters is measured by Bayesian information criteria. In this work, the speaker clustering problem is interpreted as a model selection problem, and Bayesian information criteria (BIC) is utilized to measure the similarity and as the stopping criterion. In this method, BIC is defined as the difference between two hypotheses. One hypothesis is that the pair of segments is modeled as a Gaussian, and the other one is that the each segment is separately modeled as a

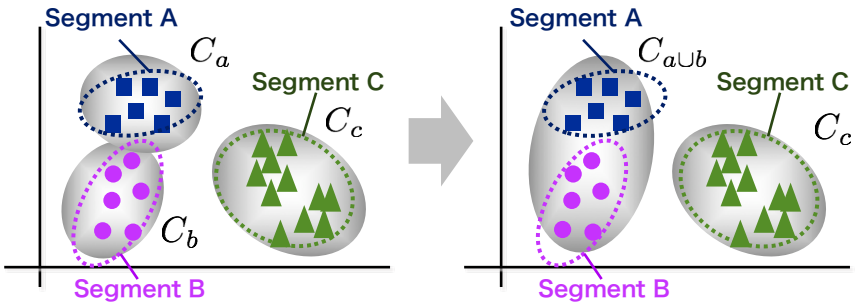


Figure 2.1: Depiction of a hierarchical agglomerative clustering (HAC) algorithm.

Gaussian. Here, BIC between clusters C_i and C_j is defined as follows:

$$BIC(C_i, C_j) = \frac{1}{2} n_i \log|\mathbf{\Sigma}_i| + \frac{1}{2} n_j \log|\mathbf{\Sigma}_j| - \frac{1}{2}(n_i + n_j)\log|\mathbf{\Sigma}| - \alpha(d + \frac{1}{2}d(d+1)) \tag{2.3}$$

where n_i and n_i respectively denote the total numbers of frames in clusters C_i and C_j, $\mathbf{\Sigma_i}$ and $\mathbf{\Sigma_i}$ respectively denote the covariance matrices of the Gaussian distributions of clusters C_i and C_j, and $\mathbf{\Sigma}$ denotes a covariance matrix of a Gaussian distribution obtained by merging them. d and α denote dimension and weight of the penalty term of BIC. The value of Eq. 2.3 is calculated for all cluster pairs, and the pair with the smallest BIC is merged. The advantage of this approach is that the obtained number of clusters is ensured to be optimal from the viewpoint of BIC. The disadvantage of this approach, on the other hand, is that the Gaussian distribution is too simple when the frame-wise observations follow a more complex distribution. Another problem of this method is that the penalty parameters are strongly dependent on the data. Moreover, some researchers have pointed out that the inappropriate clusters are often merged when the length of a segment is too short [4, 5, 6].

To represent each segment with a more flexible distribution, a mixture of Gaussian model (GMM) is introduced to represent each cluster instead of a single Gaussian component in [7, 8, 9, 10, 11]. In this case, BIC is no more analytically calculated, and various alternative criteria have been proposed to measure the similarity between GMMs, such as the cross-likelihood rate (CLR) [7], generalized likelihood ratio

(GLR) [8], KL divergence [9], and symmetric KL divergence (KL2 divergence) [10]. In the CLR-based approach [7], for example, the following value is used to measure the distance between GMMs:

$$d_{CLR}(C_A, C_B) = \frac{1}{n_A} \log \left(\frac{\mathcal{L}(\mathbf{O}_A|\mathcal{M}(\beta_A))}{\mathcal{L}(\mathbf{O}_A|\mathcal{M}(\beta_{A\cup B}))} \right) + \frac{1}{n_B} \log \left(\frac{\mathcal{L}(\mathbf{O}_B|\mathcal{M}(\beta_B))}{\mathcal{L}(\mathbf{O}_B|\mathcal{M}(\beta_{A\cup B}))} \right) \quad (2.4)$$

where n_A and n_B respectively denote the total numbers of frames in clusters C_A and C_B, and $\mathcal{L}(\mathbf{O}_A|\mathcal{M}(\beta_A))$ denotes the likelihood of segment $\mathbf{X}_A$ to cluster C_A. This CLR-based approach has achieved a high performance in many fields.

2.3 Segment-oriented generative model via mixture modeling

By introducing HAC approach, we can estimate various types of distributions for each cluster's distribution. However, this approach has at least three problems. First, the clustering accuracy seriously deteriorates if an inappropriate pair of clusters is merged. This problem becomes more serious when the number of speakers is large owing to an increase in the improper merging of clusters. This is a local solution problem caused by the lack of a procedure for dividing merged cluster [12]. Second, estimation of the distribution often fails when the number of utterances is limited. This is an overfitting problem caused by the lack of data for estimating each distribution.

In order to avoid these problems, we estimate a segment-oriented generative model as a mixture model. This estimate is almost the same as the estimate of the (frame-wise) standard mixture model, except that the observed unit is a segment. In order to describe the assignment of segments to clusters, we introduce the segment-level latent variables $\mathcal{Z} = \{z_u\}_{u=1}^{U}$. The likelihood for the set of observation vectors given the

sequence of the latent variables is expressed as follows [1]:

$$p(\mathcal{O}|\mathcal{Z},\boldsymbol{\Theta}) = \prod_{u=1}^{U}\prod_{t=1}^{T_u}\prod_{i=1}^{S} p(\mathbf{o}_{ut}|\boldsymbol{\Theta})^{\delta(z_u,i)}, \tag{2.5}$$

$$P(\mathcal{Z}|\mathbf{h}) = \prod_{u=1}^{U}\prod_{i=1}^{S} h_i^{\delta(z_u,i)}, \tag{2.6}$$

where $\delta(a,b)$ denotes the Kronecker delta, which is 1 if $a=b$ and 0 otherwise. Note that all of the frame-wise observations in the same segment are simultaneously aligned to the same distribution in this model. Since z_u denotes the index of a speaker cluster to which the u-th utterance is assigned, the speaker clustering problem is reduced to the estimation of the optimal values of the latent variables $\mathcal{Z}$ given the observed segments $\mathcal{O}$. In other words, we can obtain the optimal assignment of utterances to speaker clusters by estimating $\mathcal{Z}$ that maximizes the likelihood (ML) functions defined in Eqs. 2.5 and 2.6. This can be easily obtained by introducing the expectation maximization (EM) algorithm [13].

2.3.1 Fully Bayesian approach for segment-oriented generative model

In this case, the ML approach can suffer from overfitting of the parameters for cases with a limited number of observations $\mathcal{O}$. This causes a serious problem when it is applied to the speaker clustering problem because the number of segments is generally different from speaker to speaker. In this thesis, we introduce a fully Bayesian approach for avoiding this problem. In fully Bayesian approach, we introduce a prior distribution for model parameter to derive posterior distributions of model parameters and the latent variables. By considering a prior knowledge of model parameter (i.e. prior knowledge about the statistical property of each segment), this approach achieves a robust estimation compared with the conventional ML-based approaches. This section provides a general solution of the fully Bayesian approach for our segment-wise generative model. Using Bayes' theorem, the posterior distribution of

[1] We use the notation $p(\cdot)$ for the continuous probability function and $P(\cdot)$ for the discrete probability function.

model and latent variables is derived as follows:

$$p(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O}) \;=\; \frac{1}{H_0} p(\mathcal{O}, \mathcal{Z}|\mathbf{\Theta}) p(\mathbf{\Theta}). \tag{2.7}$$

H_0 is a normalization coefficient, which is defined as follows:

$$H_0 \triangleq p(\mathcal{O}) = \sum_{\mathcal{V},\mathcal{Z}} \int p(\mathcal{O}, \mathcal{Z}, \mathbf{\Theta}) \mathrm{d}\mathbf{\Theta}. \tag{2.8}$$

This equation, however cannot be solved analytically. Therefore, we must introduce some approximations.

Variational Bayesian (VB) approach

This section presents the VB approach and provides a general solution for VB-based model estimation. In the VB approach, the arbitrary posterior distribution $q(\mathcal{Z}, \mathbf{\Theta}, |\mathcal{O}) = q(\mathcal{Z}|\mathcal{O}) q(\mathbf{\Theta}|\mathcal{O})$ is introduced to approximate the true posterior distribution $p(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O})$ of these variables. The KL divergence between $q(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O})$ and $p(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O})$ is considered:

$$\mathrm{KL}[q(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O}) | p(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O})] = \int q(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O}) \log \frac{q(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O})}{p(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O})} \mathrm{d}\mathbf{\Theta} \tag{2.9}$$

Substituting Eq. 2.8 into Eq. 2.9 and using Jensen's inequality, the following inequality is obtained:

$$\mathrm{KL}[q(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O}) | p(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O})] \leq \log p(\mathcal{O}) - \mathcal{F}[q(\mathbf{\Theta}, \mathcal{Z}|\mathcal{O})], \tag{2.10}$$

where $\mathcal{F}[q(\mathbf{\Theta}, \mathcal{Z}|\mathcal{O})]$ is a lower bound of Eq.2.8 and defined as follows:

$$\begin{aligned} \mathcal{F}[q(\mathbf{\Theta}, \mathcal{Z}|\mathcal{O})] \;&\triangleq\; \left\langle \log \frac{p(\mathbf{\Theta}, \mathcal{Z}|\mathcal{O})}{q(\mathbf{\Theta}, \mathcal{Z}|\mathcal{O})} \right\rangle_{q(\mathbf{\Theta}, \mathcal{Z}|\mathcal{O})} \\ &=\; \left\langle \log p(\mathbf{\Theta}, \mathcal{Z}, \mathcal{O}) - \log q(\mathbf{\Theta}|\mathcal{O}) - \log q(\mathcal{Z}|\mathcal{O}) \right\rangle_{q(\mathbf{\Theta}|\mathcal{O}) q(\mathcal{Z}|\mathcal{O})} \end{aligned} \tag{2.11}$$

$\langle A \rangle_B$ denotes the expectation of A with respect to B. From Eq. 2.8, the KL divergence (i.e., statistical distance) between $q(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O})$ and $p(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O})$ becomes small as the lower bound $\mathcal{F}$ increases. Since the term $\log \mathcal{O}$ can be disregarded, the minimization of Eq. 2.8 is reduces to the maximization of $\mathcal{F}$ with respect to $q(\mathcal{Z}, \mathbf{\Theta}|\mathcal{O})$. Here, considering that $q(\mathcal{Z}, \mathbf{\Theta}, |\mathcal{O}) =$

$q(\mathcal{Z}|\mathcal{O})q(\mathbf{\Theta}|\mathcal{O})$, we obtain

$$\begin{aligned}\mathcal{F}[q(\mathbf{\Theta},\mathcal{Z}|\mathcal{O})] &\propto \left\langle\left\langle \log p(\mathbf{\Theta},\mathcal{Z},\mathcal{O})\right\rangle_{q(\mathcal{Z}|\mathcal{O})}\right\rangle_{q(\mathbf{\Theta}|\mathcal{O})} - \left\langle \log q(\mathbf{\Theta}|\mathcal{O})\right\rangle_{q(\mathbf{\Theta}|\mathcal{O})} \\ &\quad + \left\langle\left\langle \log p(\mathbf{\Theta},\mathcal{Z},\mathcal{O})\right\rangle_{q(\mathbf{\Theta}|\mathcal{O})}\right\rangle_{q(\mathcal{Z}|\mathcal{O})} - \left\langle \log q(\mathcal{Z}|\mathcal{O})\right\rangle_{q(\mathcal{Z}|\mathcal{O})} \\ &= -\mathrm{KL}\left[\left\langle \log p(\mathbf{\Theta},\mathcal{Z},\mathcal{O})\right\rangle_{q(\mathcal{Z}|\mathcal{O})} | \log p(\mathbf{\Theta}|\mathcal{O})\right] \\ &\quad -\mathrm{KL}\left[\left\langle \log p(\mathbf{\Theta},\mathcal{Z},\mathcal{O})\right\rangle_{q(\mathbf{\Theta}|\mathcal{O})} | \log p(\mathcal{Z}|\mathcal{O})\right]. \end{aligned} \tag{2.12}$$

Thus maximizing Eq. 2.12 is equivalent to minimizing the KL divergence, and the minimum occurs when $\left\langle \log p(\mathbf{\Theta},\mathcal{Z},\mathcal{O})\right\rangle_{q(\mathcal{Z}|\mathcal{O})} = \log p(\mathbf{\Theta}|\mathcal{O})$ and $\left\langle \log p(\mathbf{\Theta},\mathcal{Z},\mathcal{O})\right\rangle_{q(\mathbf{\Theta}|\mathcal{O})} = \log p(\mathcal{Z}|\mathcal{O})$. The optimal VB posterior distributions, therefore, are obtained as follows:

$$q(\mathcal{Z}|\mathcal{O}) \propto \exp\left(\left\langle \log p(\mathcal{O},\mathcal{Z},\mathbf{\Theta})\right\rangle_{q(\mathbf{\Theta})}\right), \tag{2.13}$$

$$q(\mathbf{\Theta}|\mathcal{O}) \propto \exp\left(\left\langle \log p(\mathcal{O},\mathcal{Z},\mathbf{\Theta})\right\rangle_{q(\mathcal{Z})}\right). \tag{2.14}$$

Eqs. 2.13 and 2.14 are closed-form expressions, and they are estimated by the expectation and Maximization (EM) algorithm.

Problems with VB approach

This VB-based procedure monotonically increases the free energy, as described in Eq. 2.11 under the variational posterior distribution $q(\mathcal{Z},\mathbf{\Theta})$, but this approach suffers from two problems caused by the difference between the true and variational posterior distributions as well as the biased values. The first problem is that the true posterior distributions of the latent variable and the model parameters in a segment-oriented generative model cannot be factorized (i.e., $p(\mathbf{\Theta},\mathcal{Z}|\mathcal{O} \neq p(\mathcal{Z}|\mathcal{O})\ p(\mathbf{\Theta}|\mathcal{O})$), although the variational posterior distributions assume that they can. The second problem is that the posterior probability obtained is generally biased because the calculated statistics are strongly biased by the size of each segment. These problems are especially severe when the number of segments is limited. To solve these problems, we need to

estimate the marginalized posterior distributions, into which model parameter $\boldsymbol{\Theta}$ are collapsed[1]. This is obtained by marginalizing Eq. 2.7 with respect to these parameters as follows:

$$P(\mathcal{Z}|\mathcal{O}) \;=\; \frac{1}{H_0}\int p(\mathcal{Z},\boldsymbol{\Theta},\mathcal{O})\mathrm{d}\boldsymbol{\Theta}. \tag{2.15}$$

We can then estimate the posterior distribution of the latent variables directly and obtain an unbiased estimation.

Collapsed VB methods for estimating the marginalized posterior distribution have been proposed in several studies [14, 15], but these approaches are generally infeasible for our hierarchical model because we cannot apply the approximation of convexity to a hierarchical structure. Therefore, we introduce the Markov chain Monte Carlo (MCMC) method to estimate the marginalized posterior distribution from Eq. 2.15.

Markov chain Monte Carlo (MCMC) approach with Collapsed Gibbs sampler

The goal of the MCMC approach is to obtain samples from Eq. 2.15. We can derive the marginalized distribution with respect to the model parameters described in Eq. 2.15 because we do not need to evaluate the normalization term Eq. 2.15 when employing an MCMC approach. In this thesis, we select a collapsed Gibbs sampling because of its simplicity. In each step of the collapsed Gibbs sampling process, the value of one of the latent variables (e.g., z_u) is replaced with a value generated from the distribution of that variable given the values of the remaining latent variables (i.e., $\mathcal{Z}^*_{\backslash u} = \{z_{u'}|u' \neq u\}$). In the case of our segment-oriented generative model, we iteratively sample from following conditional posterior

$$z_u \sim P(z_u|\mathcal{O},\mathcal{Z}^*_{\backslash u}). \tag{2.16}$$

The optimal values of $\mathcal{Z}$ (i.e., the optimal assignments of utterances to clusters) can be obtained from its posterior distribution $P(\mathcal{Z}|\mathcal{O})$ by iterating to sample z_u from its conditional posterior distribution in Eq. 2.16 until convergence.

[1] In this case, "collapsed" means that samples are drawn from the marginalized distribution with respect to the model parameter $\boldsymbol{\Theta}$. In the following, we refer to collapsed Gibbs sampling simply as Gibbs sampling.

Relationship with Hidden Markov model-based approach

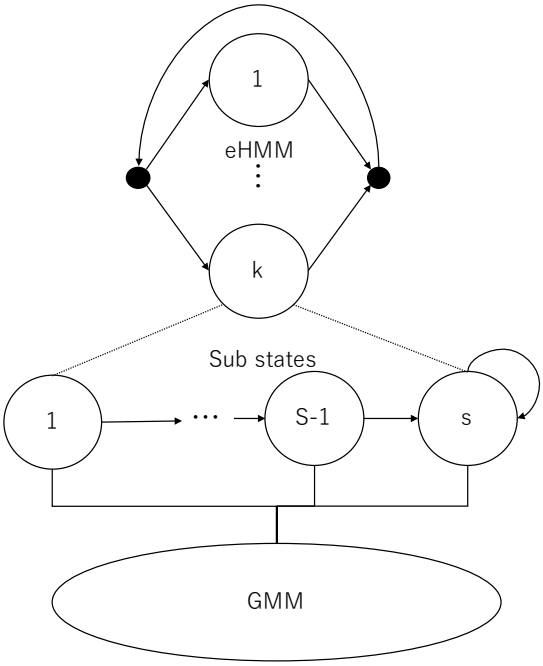


Figure 2.2: An ergodic HMM topology used for clustering in [1].

Our segment-oriented generative model approach is closely related to a model-based clustering based on an ergodic Hidden Markov Model (eHMM) [1]. In this model, each state of eHMM has S sub-states and all of these sub-states share a GMM as shown in 2.2. In this approach, the optimal number of states (i.e. the number of clusters) by merging the states with respect to BIC. [1] extended this approach in order to avoid overly frequent transitions by constraining the duration period of the same speaker. These approaches explicitly model the speaking time of each speaker using state transition probabilities. [16] applied an ergodic hidden Markov model (HMM), which is a fully connected HMM where all possible transitions are allowed, to speaker clustering. Here, each state of an HMM corresponds to a speaker cluster. [1, 17] extended this approach so as to avoid overly frequent transitions by constraining the duration period of the same speaker. These approaches explicitly model the speaking time of each speaker using state transition probabilities. In contrast, our segment-oriented generative model assumes that the boundaries of speaker clusters can be detected by some means.

In fact, these boundaries can be determined using existing voice activity detection (VAD) methods, which have been successfully applied to automatic speech recognition and noise reduction applications. In this thesis, therefore, we defined the model with an explicit boundary.

2.3.2 Mixture-of-mixtures modeling

In the segment-oriented generative modeling defined as Eq. 2.1, the type of distribution of each component $p(\mathbf{O}_u|\mathbf{\Theta}_i)$ has a crucial effect on model performance. Mixture models are reasonable approximations for representing inner-data variability [18, 19] and various distributions have been used as components of mixture models such as the t-distribution [20] and von Mises-Fisher distribution [21, 22]. In particular, Gaussian distributions are used widely as a reasonable approximations for a wide class of probability distributions [23]. By using a mixture distribution to represent each cluster, the segment generative model is modeled as a mixture of these mixture distributions. We refer to this as a *mixture-of-mixture* model. The optimal assignment of higher-level observations to clusters can be obtained by evaluating the posterior probability of assigning each observation to each cluster's mixture distribution. Thus, the clustering problem is reduced to the problem of estimating this mixture-of-mixture model.

Related works of mixture-of-mixtures modeling

The concept of mixture-of-mixture modeling was firstly introduced to analyze multi-modal data sample observations comprising both continuous and categorical variables [24, 25]. Mixture-of-mixture modeling also have been applied to data that composed of sets of observations such as data from students nested within schools or patients within hospitals [26, 27, 28]. These approaches have been proven to be effective for estimating a hierarchical structure from hierarchically structured data. All of Gaussian component, for example, share the same scaling parameter in the approach of [29] and ones in the same class shares the scaling parameter in the approach of [30]. The main advantage of introducing mixture distributions for cluster representation is that we can model

inner-cluster variability in each cluster by the cluster specific mixture distribution. However, in these studies, the applications of mixture-of-mixture modeling were limited to simulated or low-dimensional data.

When we apply a statistical model to high-dimensional data, we often suffer from a over-fitting problem. This problem is more serious in the estimation of mixture-of-mixtures model, because this model generally has a large number of parameters than a mixture model composed of single distributions. In the case where Gaussian mixture models (GMMs) is introduced as component distributions of mixture-of-mixture models, for example, the number of parameters is $O(S \times M \times D^2)$, which grows linearly in terms of the number of clusters S and the number of mixtures M, but quadratically in terms of the number of dimensions D. where S, M and D are the number of clusters, the number of mixtures in each cluster and dimensions, respectively. $S \times M \times D(\frac{D}{2} + 1) + S(M + 1)$ The number of parameters still remains $O(S \times M \times D)$, $S \times M \times 2D + S(M + 1)$ even when we use a spherical Gaussian distribution or diagonal-covariance Gaussian distribution. and therefore we require large number of data to estimate these model parameters. Some studies solve this problem by introducing factor analysis to this model structure [26]. When the number of parameters to be estimated is large, the shortage of the data makes the system less reliable. in relation to the number of dimensions. A robust model estimation method, therefore, is required to estimate the structure of mixture-of-mixtures model from high-dimensional data.

In [26, 27, 28], an expectation maximization (EM) approach [13] was used to estimate mixture-of-mixture models by augmenting observations with two-level (higher-level and lower-level) latent variables. However, this maximum likelihood-based approach often suffers from an over-fitting problem when applied to high-dimensional data. This problem becomes more serious when the amount of data is limited [31, 32]. In this thesis, therefore, we alternative approach based on fully Bayesian approach for this mixture-of-mixture model.

2.4 Summary

This chapter provided a detailed definition of a segment-oriented generative model. We reviewed HAC approaches based on both single and mixture distributions, and we explained that HAC approaches have intrinsic drawbacks caused by their deterministic procedures. We then showed an alternative approach which estimates segment-oriented generative model as a mixture model. We formalized this mixture model with a fully Bayesian approach to realize robust estimation against the shortage of data. We also showed that the mixture-of-mixtures modeling is derived by introducing mixture distribution to each cluster's generative model.

Chapter 3

Segment-oriented generative clustering via a single distribution

3.1 Introduction

The aim of this chapter is to show a superiority of a segment-oriented clustering over conventional HAC-based methods. In this section, all cluster in our segment-oriented mixture model is modeled by a Gaussian distribution in order to show the potential of the sampling-based model estimation for segment-oriented mixture model. We also attempt to develop a segment-oriented generative technique able to estimate the number of speakers by employing a nonparametric Bayesian framework [33]. Here, we derive the segment-oriented speaker mixture model for infinite speakers by simply taking the limit of the formula of the finite speaker mixture model as the number of speakers approaches infinity. We call this model the segment-oriented Dirichlet process mixture model (SO-DPMM). This chapter demonstrates that SO-DPMM achieves more accurate speaker clustering than the HAC-based system developed under the same condition and can cope with practical large-scale data.

3.2 Segment-oriented mixture model for finite speakers

First, we derive the segment-oriented mixture model when the number of speaker clusters is fixed. Here, we describe how to estimate the segment-oriented generative model for the finite speakers and how to assign speaker labels to each segment by using this model.

3.2.1 Segment-oriented mixture model

Here, let us redefine segment-oriented generative model using a single Gaussian distribution.

We assume that a D-dimensional Gaussian distribution for each speaker generates the segments from the corresponding speaker and that the variability for all speakers is modeled by a mixture of these distributions (i.e., a Gaussian mixture model (GMM)). We then assume that each segment is generated as an i.i.d. from this GMM and that each feature vector $\mathbf{o}_{ut}$ is generated as an i.i.d. from a mixture component to which the segment is assigned. $\mathcal{Z} = \{z_u\}_{u=1}^{U}$ represents the indexes of speaker clusters. In this segment-oriented mixture model, the likelihood for the set of observation vectors given the sequence of the latent variables is expressed as follows:

$$p(\mathcal{O}|\mathcal{Z}, \mathbf{\Theta}) = \prod_{u=1}^{U}\prod_{i=1}^{S}\prod_{t=1}^{T_u}\mathcal{N}(\mathbf{o}_{ut}|\boldsymbol{\mu}_i, \mathbf{\Sigma}_i)^{\delta(z_u,i)}, \tag{3.1}$$

$$P(\mathcal{Z}|\mathbf{h}) = \prod_{u=1}^{U}\prod_{i=1}^{S} h_i^{\delta(z_u,i)}, \tag{3.2}$$

where $\delta(a, b)$ denotes the Kronecker delta, which is 1 if $a = b$ and 0 otherwise. $\mathbf{h} = \{h_i\}_{i=1}^{S}$ and $\mathbf{\Theta} = \{\boldsymbol{\mu}_i, \mathbf{\Sigma}_i\}_{i=1}^{S}$ denote the set of weights, mean vector, and covariance matrix for S speaker clusters, respectively. $\mathbf{\Sigma}_i$ is a diagonal covariance matrix whose (d, d)-th element is represented by $\sigma_{i,dd}$.

3.2.2 Fully Bayesian approach for segment-oriented mixture model

To derive the Bayesian representation, we need to introduce the conjugate prior distributions of the model parameters $\mathbf{\Theta}$. In the case where each distribution is modeled as a Gaussian distribution, the following conjugate prior is introduced:

$$p(\mathbf{\Theta}, \mathbf{h}) = \begin{cases} \{h_i\}_{i=1}^{S} & \sim \mathcal{D}(\mathbf{h}^0) \\ \{\mu_{i,d}, \Sigma_{i,d}\}_{i=1}^{S} & \sim \prod_d \mathcal{N}G(\mu_d^0, \xi^0, \eta^0, \sigma_{dd}^0), \forall i \end{cases} \tag{3.3}$$

where $\mathcal{D}(\mathbf{h}^0)$ denotes the Dirichlet distribution with a hyper parameter $\mathbf{h}^0 = \{h_0/S, \cdots, h_0/S\}$ and $\mathcal{NG}(\mu_d^0, \xi^0, \eta^0, \sigma_{dd}^0)$ denotes the Gaussian-Gamma distribution with hyper parameters μ_d^0, ξ^0, η^0, and σ_{dd}^0. Note that these hyper-parameters do not depend on each cluster [1]. The graphical model for this model is shown in Fig. 3.1 (a). By using these prior distributions, we can derive the joint distribution for the complete data case.

Marginalized likelihood for the complete data case

For the complete data case, the posterior probabilities of the latent variables, $P(\mathcal{Z}|\mathcal{O})$, return 0 or 1 because all assignments of segments to speaker clusters are available. Then, the sufficient statistics of this model can be described as follows:

$$\begin{cases} n_i^{\text{utt}} & = \sum_u \delta(z_u, i) \\ n_i^{\text{frm}} & = \sum_u \delta(z_u, i) T_u \\ \mathbf{m}_i & = \sum_u \delta(z_u, i) \sum_t \mathbf{o}_{ut} \\ r_{i,dd} & = \sum_u \delta(z_u, i) \sum_t (o_{ut,d})^2 \end{cases} \tag{3.4}$$

where n_i^{utt} and n_i^{frm} are the number of segments and that of frames assigned to the i-th cluster, respectively; $\mathbf{m}_i$ and $r_{i,dd}$ are the first- and second-order sufficient statistics, respectively. By using Eqs. 3.2 and 3.4, the likelihood for the complete data case can be expressed as follows:

$$p(\mathcal{O}, \mathcal{Z}|\mathbf{\Theta}, \mathbf{h}) = \prod_i (h_i)^{n_i^{\text{utt}}} \prod_{u,t} \mathcal{N}(\mathbf{o}_{ut}|\boldsymbol{\mu}_i, \mathbf{\Sigma}_i)^{\delta(z_u,i)}. \tag{3.5}$$

Here, recalling that the speaker clustering problem aims to estimate the optimal assignment of segments to speaker clusters, we can see that the parameter $\mathbf{\Theta}$ need not be estimated. We can therefore marginalize this parameter out from the joint distribution described in Eq. 3.5. This marginalization allows us to optimize the model on the latent variable space. By restricting the search space of the latent variables, we can obtain a model estimation algorithm that is robust against the local optima problem.

From Eqs. 3.3 and 3.5, the marginalized likelihood for the complete data case, integrated by using the parameter $\mathbf{\Theta}$, can be factorized to the

[1] The detailed definition of Dirichlet and Gaussian-Gamma distributions is described in Appendix A.1.2

following two integrals:

$$\begin{aligned} p(\mathcal{O}, \mathcal{Z}) &= \int p(\mathcal{O}, \mathcal{Z}|\mathbf{\Theta}, \mathbf{h}) \cdot p(\mathbf{\Theta}, \mathbf{h}) \mathrm{d}\mathbf{\Theta} \mathrm{d}\mathbf{h} \\ &= \int P(\mathcal{Z}|\mathbf{h}) p(\mathbf{h}) \mathrm{d}\mathbf{h} \cdot \int p(\mathcal{O}|\mathcal{Z}, \mathbf{\Theta}) p(\mathbf{\Theta}) \mathrm{d}\mathbf{\Theta}. \end{aligned} \tag{3.6}$$

The first term on the right-hand side of Eq. 3.6 is described as follows:

$$\int P(\mathcal{Z}|\mathbf{h}) p(\mathbf{h}) \mathrm{d}\mathbf{h} = C(\mathbf{h}^0) \frac{\prod_i \Gamma(\tilde{h}_i)}{\Gamma(\sum_i \tilde{h}_i)}, \tag{3.7}$$

where $C(\mathbf{h}^0)$ denotes the normalization term that is independent of n_i^{utt}. The second term on the right-hand side of Eq. 3.6 is described as follows:

$$\begin{aligned} &\int p(\mathcal{O}|\mathcal{Z}, \mathbf{\Theta}) p(\mathbf{\Theta}) \mathrm{d}\mathbf{\Theta} \\ &= \prod_i (2\pi)^{-\frac{n_i^{\text{frm}} D}{2}} \frac{(\xi^0)^{\frac{D}{2}} \left(\Gamma\left(\frac{\eta_i^0}{2}\right)\right)^{-D} \left(\prod_d \sigma_{i,dd}^0\right)^{\frac{\eta_i^0}{2}}}{(\tilde{\xi}_i)^{\frac{D}{2}} \left(\Gamma\left(\frac{\tilde{\eta}_i}{2}\right)\right)^{-D} \left(\prod_d \tilde{\sigma}_{i,dd}\right)^{\frac{\tilde{\eta}_i}{2}}} \\ &= \prod_i \frac{Z(\tilde{\xi}_i, \tilde{\eta}_i, \tilde{\boldsymbol{\mu}}_i, \tilde{\mathbf{\Sigma}}_i)}{Z(\xi^0, \eta^0, \boldsymbol{\mu}^0, \mathbf{\Sigma}^0)} (2\pi)^{-\frac{n_i D}{2}}, \end{aligned} \tag{3.8}$$

where $\tilde{\mathbf{\Theta}} \triangleq \left\{\tilde{h}_i, \tilde{\eta}_{i,dd}, \tilde{\xi}_{i,dd}, \tilde{\boldsymbol{\mu}}_i, \tilde{\sigma}_{i,dd}\right\}$ denotes the hyper-parameter of the posterior distribution for $\mathbf{\Theta}$, which is described as follows:

$$\begin{cases} \tilde{h}_i & = h_i^0 + n_i^{\text{utt}} \\ \tilde{\xi}_i & = \xi^0 + n_i^{\text{frm}} \\ \tilde{\eta}_i & = \eta^0 + n_i^{\text{frm}} \\ \tilde{\boldsymbol{\mu}}_i & = \frac{\xi^0 \boldsymbol{\mu}_i^0 + \mathbf{m}_i}{\tilde{\xi}_i} \\ \tilde{\sigma}_{i,dd} & = \sigma_{i,dd}^0 + r_{i,dd} + \xi^0 (\mu_{i,d}^0)^2 - \tilde{\xi}_i (\tilde{\mu}_{i,d})^2 \end{cases} \tag{3.9}$$

where we have used Eq. 3.4.

MCMC-based posterior estimation

We again emphasize that the speaker clustering problem is reduced to the estimation of the latent variables $\mathcal{Z}$, which maximize the posterior distribution $P(\mathcal{Z}|\mathcal{O})$. We can then derive the posterior distribution for the latent variables as $p(\mathcal{Z}|\mathcal{O}) \propto p(\mathcal{O}, \mathcal{Z})$. The evaluation of all combinations of these latent variables in $p(\mathcal{Z}|\mathcal{O})$, however, is obviously infeasible if the number of segments (i.e. the number of latent variables) is large.

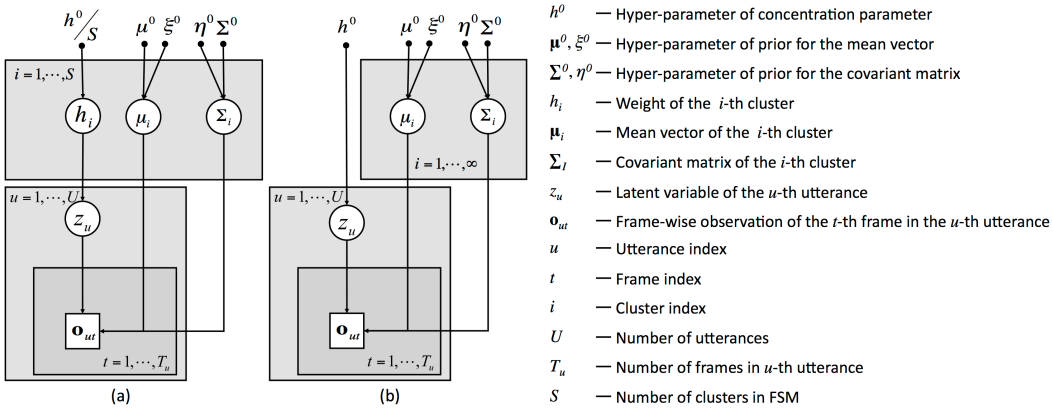


Figure 3.1: Graphical models of segments-oriented mixture models for (a) finite and (b) infinite speakers.

Instead, we use collapsed Gibbs sampling [34] to obtain the optimal value of $\mathcal{Z}$ directly from its posterior distribution $P(\mathcal{Z}|\mathcal{O})$.

In each step of the collapsed Gibbs sampling process, the value of one of the latent variables (e.g., z_u) is replaced with a value generated from the distribution of that variable given the values of the remaining latent variables (i.e., $\mathcal{Z}^*_{\backslash u} = \{z_{u'}|u' \neq u\}$). In this case, the latent variables are sampled from the conditional posterior distribution as follows:

$$\begin{aligned} & P(z_u = i'|\mathcal{O}, \mathcal{Z}^*_{\backslash u}) \\ & \propto P(z_u = i'|\mathcal{Z}^*_{\backslash u}) \cdot p(\mathcal{O}_u|\mathcal{O}_{\backslash u}, \mathcal{Z}^*_{\backslash u}, z_u = i') \\ & = \frac{P(\mathcal{Z}^*_{\backslash u}, z_u = i')}{P(\mathcal{Z}^*_{\backslash u})} \cdot \frac{p(\mathcal{O}_u, \mathcal{O}_{\backslash u}|\mathcal{Z}^*_{\backslash u}, z_u = i')}{p(\mathcal{O}_{\backslash u}|\mathcal{Z}^*_{\backslash u})}. \end{aligned} \tag{3.10}$$

Note that the hyper-parameters of prior distributions, $\{\mathbf{h}^0, \boldsymbol{\Theta}^0\}$, are omitted in Eq. 3.10. From Eq. 3.7, the first term on the right-hand side of Eq. 3.10 can be described as follows:

$$\frac{P(\mathcal{Z}^*_{\backslash u}, z_u = i')}{P(\mathcal{Z}^*_{\backslash u})} = \frac{\frac{h^0}{S} + n_{i'}}{U - 1 + h^0}. \tag{3.11}$$

From Eq. 3.8, the second term on the right-hand side of Eq. 3.10 is described as follows:

$$\frac{p(\mathcal{O}|\mathcal{Z}^*_{\backslash u}, z_u = i')}{p(\mathcal{O}_{\backslash u}|\mathcal{Z}^*_{\backslash u})} \propto \exp\left(g_{i'}(\tilde{\boldsymbol{\Theta}}_{i'}) - g_{i'}(\tilde{\boldsymbol{\Theta}}_{i'\backslash u})\right), \tag{3.12}$$

where

$$\begin{aligned} g_{i'}(\tilde{\boldsymbol{\Theta}}_{i'}) & \triangleq \ln p(\mathcal{O}|\mathcal{Z}^*_{\backslash u}, z_u = i') \\ & = D \log \Gamma\left(\frac{\tilde{\eta}_{i'}}{2}\right) - \frac{D}{2} \log \tilde{\xi}_{i'} - \frac{\tilde{\eta}_{i'}}{2} \sum\nolimits_d \log \tilde{\sigma}_{i',dd} \end{aligned} \tag{3.13}$$

$$
\begin{aligned}
g_{i'}(\tilde{\boldsymbol{\Theta}}_{i'\backslash u}) \quad &\triangleq \ln p(\mathcal{O}_{\backslash u}|\mathcal{Z}^*_{\backslash u}) \\
&= D\log\Gamma\left(\frac{\tilde{\eta}_{i'\backslash u}}{2}\right) - \frac{D}{2}\log\tilde{\xi}_{i'\backslash u} - \frac{\tilde{\eta}_{i'\backslash u}}{2}\sum\nolimits_d \log\tilde{\sigma}_{i'\backslash u,dd}.
\end{aligned} \tag{3.14}
$$

$\tilde{\boldsymbol{\Theta}}_{i'\backslash u}$ in Eq. 3.14 denotes the hyper-parameter of the posterior distribution for $\boldsymbol{\Theta}$ after removing u-th segments, which is described as follows:

$$
\tilde{\boldsymbol{\Theta}}_{i'\backslash u} \triangleq \left\{
\begin{array}{ll}
\tilde{\xi}_{i'\backslash u} & = \tilde{\xi}_i - T_u \\
\tilde{\eta}_{i'\backslash u} & = \tilde{\eta}_i - T_u \\
\tilde{\boldsymbol{\mu}}_{i'\backslash u} & = \frac{\tilde{\xi}_{i'}\tilde{\boldsymbol{\mu}}_{i'} - \sum_t \boldsymbol{o}_{ut}}{\tilde{\xi}_{i'\backslash u}} \\
\tilde{\sigma}_{i'\backslash u,dd} & = \sigma^0_{i',dd} + r_{i',dd} - \sum_t (o_{ut,d})^2 \\
 & \quad + \xi^0(\mu^0_{i',d})^2 - \tilde{\xi}_{i'\backslash u}(\tilde{\mu}_{i'\backslash u,d})^2
\end{array}
\right. \tag{3.15}
$$

The optimal values of $\mathcal{Z}$ (i.e., the optimal assignments of segments to clusters) can be obtained from its posterior distribution $P(\mathcal{Z}|\mathcal{O})$ by iterating to sample z_u from its conditional posterior distribution in Eq. 3.10 until convergence.

3.3 Segment-oriented mixture model for infinite speakers

This section attempt to extend the segment-oriented generative model for finite speakers in order to deal with infinite speakers. For this purpose, we introduce Dirichlet process as the prior distribution of mixture weights. The derived model is a type of Dirichlet process mixture model (DPMM) [33], but it differs from the original DPMM in that the generative unit is not a frame but rather an segments. In this research we call this model "segment-oriented Dirichlet process mixture model (SO-DPMM)." In the present study, SO-DPMM is built by using Chinese restaurant process (CRP) [35], which can avoid local solutions because of its sampling-based implementation. Furthermore, we can easily integrate CRP with other sophisticated methods, such as simulated annealing. The graphical model of the segment-oriented mixture model for infinite speakers is shown in Fig. 3.1 (b). Table 1 provides a pseudo code of this method.

CRP is found by taking the limit of S (i.e., $S \to \infty$) in Eq. 3.10. Note that there are at most $U(< S)$ speaker clusters to which at least

one segment is assigned. In the case of S being infinite, most clusters should be empty. In this case, we can separately compute Eq. 3.11 for the case where the u-th segment is assigned to a cluster with more than one segment (i.e., $n_{i'} > 0$) and the case where the u-th segment is assigned to a new cluster with no segment (i.e., $n_{i'} = 0$).

$$\frac{P(\mathcal{Z}^*_{\backslash u}, z_u = i')}{P(\mathcal{Z}^*_{\backslash u})} = \begin{cases} \frac{\frac{h^0}{S}+n_{i'}}{U-1+h^0}, & \text{if } i' = z_k \text{ for } \exists k \neq u \\ \frac{\frac{h^0}{S}}{U-1+h^0}, & \text{if } i' \neq z_k \text{ for } \forall k \neq u \end{cases} \tag{3.16}$$

By taking the limit of $S \to \infty$, the number of segments U satisfies $U \ll S$ and thus we can assume that there are S empty clusters. Therefore, by combining the empty clusters, Eq. 3.16 is described as follows:

$$\frac{P(\mathcal{Z}^*_{\backslash u}, z_u = i')}{P(\mathcal{Z}^*_{\backslash u})} = \begin{cases} \frac{\frac{h^0}{S}+n_{i'}}{U-1+h^0}, & \text{if } i' = z_k \text{ for } \exists k \neq u \\ S \cdot \frac{\frac{h^0}{S}}{U-1+h^0}, & \text{if } i' \neq z_k \text{ for } \forall k \neq u \end{cases} \tag{3.17}$$

Taking the limit of $S \to \infty$ in Eq. 3.17 allows us to derive the following equation:

$$\frac{P(\mathcal{Z}^*_{\backslash u}, z_u = i')}{P(\mathcal{Z}^*_{\backslash u})} = \begin{cases} \frac{n_{i'}}{U-1+h^0}, & \text{if } i' = z_k \text{ for } \exists k \neq u \\ \frac{h^0}{U-1+h^0}, & \text{if } i' \neq z_k \text{ for } \forall k \neq u \end{cases} \tag{3.18}$$

From Eq. 3.8, we can also separately compute the second term on the right-hand side of Eq. 3.10 as follows:

$$\frac{p(\mathcal{O}, \mathcal{Z}^*_{\backslash u}, z_u = i')}{p(\mathcal{O}_{\backslash u}, \mathcal{Z}^*_{\backslash u})} = \begin{cases} \exp\left(g_{i'}(\tilde{\boldsymbol{\Theta}}_{i'}) - g_{i'}(\tilde{\boldsymbol{\Theta}}_{i'\backslash u})\right), \\ \qquad \text{if } z_k = i' \text{ for } \exists k \neq u \\ \exp\left(g_{\text{new}}(\tilde{\boldsymbol{\Theta}}_{\text{new}}) - g_{\text{new}}(\boldsymbol{\Theta}_0)\right) \\ \qquad \text{if } z_k \neq i' \text{ for } \forall k \neq u \end{cases} \tag{3.19}$$

where $g_{\text{new}}(\tilde{\boldsymbol{\Theta}}_{\text{new}})$ and $g_{\text{new}}(\boldsymbol{\Theta}_0)$ denote the logarithmic likelihood for $\mathcal{O}_u$ to the new cluster, and the prior likelihood of the parameter itself, respectively.

We can evaluate both $g_{\text{new}}(\tilde{\boldsymbol{\Theta}}_{\text{new}})$ and $g_{\text{new}}(\boldsymbol{\Theta}_0)$ by using Eq. 3.13, noting that only the u-th segment is assigned to the new cluster for $g_{\text{new}}(\tilde{\boldsymbol{\Theta}}_{\text{new}})$ and no ones are assigned to the new cluster for $g_{\text{new}}(\boldsymbol{\Theta}_0)$. That is, we can respectively evaluate $g_{\text{new}}(\boldsymbol{\Theta}_0)$ and $g_{\text{new}}(\tilde{\boldsymbol{\Theta}}_{\text{new}})$ by substituting $\tilde{\boldsymbol{\Theta}}_{i'}$ in Eq. 3.13 to $\boldsymbol{\Theta}_0$ and $\tilde{\boldsymbol{\Theta}}_{\text{new}}$, which is described as follows:

Algorithm 1: Speaker clustering by using SO-DPMM. Threshold is 30 for TIMIT and 50 for CSJ.

Initialize S and $\{z_u\}_{u=1}^{U}$.
repeat
  for all u = shuffle $(1, \cdots, U)$ **do**
    Sample z_u according to Eq. 3.21.
    if $z_u = S + 1$ **then**
      $\boldsymbol{\Theta}_{S+1} \sim G_0\left(\boldsymbol{\Theta}|\boldsymbol{\Theta}^0\right)$.
      $S \leftarrow S + 1$.
    end if
  end for
until number of iterations exceeds threshold

$$\tilde{\boldsymbol{\Theta}}_{\text{new}} \triangleq \begin{cases} \tilde{\xi}_{\text{new}} & = \xi_0 + T_u \\ \tilde{\eta}_{\text{new}} & = \eta_0 + T_u \\ \tilde{\boldsymbol{\mu}}_{\text{new}} & = \frac{\boldsymbol{\mu}_0 + \sum_t \boldsymbol{o}_{ut}}{\tilde{\xi}_{\text{new}}} \\ \tilde{\sigma}_{new,dd} & = \sigma_{dd}^0 + \sum_t (o_{ut,d})^2 \\ & \quad + \xi^0 (\mu_d^0)^2 - \tilde{\xi}_{\text{new}} (\tilde{\mu}_{\text{new},d})^2 \end{cases} \tag{3.20}$$

From Eqs. 3.18 and 3.19, the posterior probability of the latent variables can be finally described as follows:

$$p(z_u = i' | \mathcal{O}, \mathcal{Z}_{\backslash u}) \quad \propto \begin{cases} \frac{n_{i'}}{U-1+h^0} \cdot \exp\left(g_i(\tilde{\boldsymbol{\Theta}}_{i'}) - g_i(\tilde{\boldsymbol{\Theta}}_{i'\backslash u}) \right), \\ \qquad \text{if } z_k = i' \text{ for } \exists k \neq u \\ \frac{h^0}{U-1+h^0} \cdot \exp\left(g_{\text{new}}(\tilde{\boldsymbol{\Theta}}_{\text{new}}) - g_{\text{new}}(\boldsymbol{\Theta}_0) \right) \\ \qquad \text{if } z_k \neq i' \text{ for } \forall k \neq u \end{cases} \tag{3.21}$$

We iteratively reassign each segment to one of the existing clusters or the new cluster in proportion to Eq. 3.21 until the value of the samples converges. As shown in Eq. 3.21, the hyper-parameter h^0 determines how frequently each segment is reassigned to the new cluster. The estimated number of speaker clusters, therefore, depends on the value of h^0. In the next section, we demonstrate that this parameter can be tuned by using a development set.

3.4 Speaker clustering experiments

We carried out the speaker clustering experiments by using the TIMIT [36] and Corpus of Spontaneous Japanese (CSJ) [37] databases. We compared SO-DPMM described in Section 3.3 with existing hierarchical agglomerative clustering based on the Bayesian information criterion (HAC-BIC) [4] in speaker clustering with estimation of the number of speakers.

HAC-BIC is similar to SO-DPMM in terms of the model structure, i.e., both methods assume that each speaker can be represented by a single Gaussian and estimate the number of clusters using model complexity. Here, the aim of the present study is to verify if the model-based speaker clustering approach can be extended so as to estimate the number of speakers by incorporating the nonparametric Bayesian techniques in the segment-oriented speaker mixture model. We therefore are determined to focus on comparing SO-DPMM and HAC-BIC to make this experiment comparable.

Another agglomerative approach is HAC based on the Gaussian mixture model (HAC-GMM) and this is one of the state-of-the-art methods. The aim of this chapter, however, is to verify the effectiveness of the model-based speaker clustering for the infinite speakers over the hierarchical agglomerative clustering. As aforementioned, we are determined to compare both methods under the condition of a single Gaussian being used as a speaker distribution to make the both frameworks comparable.

3.4.1 Speech data

We performed the speaker clustering experiments by using six evaluation sets obtained from the TIMIT and CSJ databases. We used two evaluation sets in TIMIT. T-1 was the “core test set,” which included 192 segments spoken by 24 speakers. T-2 was the “complete test set,” which excluded the core test set in the TIMIT database and included 1,152 segments spoken by 144 speakers. T-1 and T-2 are balanced data, in which each speaker spoke the same number of segments. The remaining four evaluation sets were obtained from lectures in CSJ as follows.

Table 3.1: Details of test set. # speakers, # segments, # samples, and total duration denote the number of speakers, number of segments, number of frame-wise observations, and total duration.

	T-1	T-2	C-1	C-2	C-3	C-4
# speakers	24	144	5	10	5	10
# segments (# samples)	192 (5.8 K)	1,152 (353 K)	500 (209 K)	1,000 (404 K)	9,333 (4.0 M)	15,435 (6.4 M)
total duration	9.7 [min.]	59.0 [min.]	35.0 [min.]	1.1 [h]	11.1 [h]	17.6 [h]

First, all lectures were divided into segment units based on the segments of silence in their transcriptions that were longer than 500 ms; 5 and 10 speakers were then randomly selected and their 100 segments were selected for C-1 and C-2. Each segment was between 2 and 10 s long. Next, we selected another 5 and 10 speakers and all their segments for C-3 and C-4. C-3 and C-4 are "unbalanced" and large-scale data (they include approximately 4 and 6 million samples, respectively). Table 3.1 lists the number of speakers and segments in the evaluation sets used. Speech data were sampled at 16 kHz and quantized into 16-bit data.

We used 12-dimensional mel-frequency cepstrum coefficients (MFCCs) as the feature parameters. The frame length and shift were 25 ms and 10 ms, respectively.

3.4.2 Speaker clustering experiments in which the number of speakers is known

We attempted to seek a suitable optimization in the segment-oriented mixture model under the finite speaker condition when the correct number of speakers is given. In this experiment, we evaluated Gibbs sampling-based, maximum likelihood-based, and BIC-based methods.

Experimental conditions

The hyper-parameters in Eq. 4.6 were set as follows: $h^0 = 1$, $\xi^0 = 1$ and $\eta^0 = 1$. $\boldsymbol{\mu}_i^0$ and $\boldsymbol{\Sigma}_i^0$ were computed as the mean and covariance of all data used in the database.

Table 3.2: Speaker clustering results for TIMIT. ACP, ASP and K represent average cluster purity, average speaker purity and their geometric mean, respectively. Note that the number of clusters is given in this experiment.

Eval.	Method	ACP	ASP	K
T-1	Gibbs	0.76	0.81	0.79
(spkr:24)	ML-EM	0.63	0.74	0.68
(utt:192)	HAC-BIC	0.78	0.81	**0.80**
T-2	Gibbs	0.51	0.53	**0.52**
(spkr:144)	ML-EM	0.33	0.47	0.39
(utt:1,152)	HAC-BIC	0.50	0.52	0.51

Table 3.3: Speaker clustering results for CSJ. DER represents the speaker diarization error rate Note that the number of clusters is given in this experiment.

Eval.	Method	ACP	ASP	K	DER [%]
C-1	Gibbs	0.87	0.94	**0.91**	**0.11**
(spkr:5)	ML-EM	0.85	0.94	0.89	0.13
(utt:500)	HAC-BIC	0.79	0.76	0.77	0.38
C-2	Gibbs	0.78	0.88	0.83	0.19
(spkr:10)	ML-EM	0.82	0.89	**0.86**	**0.15**
(utt:1,000)	HAC-BIC	0.58	0.77	0.67	0.39
C-3	Gibbs	0.86	0.88	**0.87**	**0.13**
(spkr:5)	ML-EM	0.84	0.88	0.86	0.19
(utt:9,333)	HAC-BIC	0.24	0.25	0.24	0.71
C-4	Gibbs	0.83	0.83	**0.83**	**0.14**
(spkr:10)	ML-EM	0.80	0.80	0.80	0.22
(utt:15,435)	BIC	0.24	0.25	0.24	0.71

In this experiment, we supposed that the optimal number of clusters is known and that this is fixed to its true value. We then applied the Gibbs sampling algorithm described in section 3.2 (**Gibbs**). Furthermore, in order to allow a comparison, we performed the EM algorithm, which maximizes the likelihood for the complete data case, as described in Eq. 3.5 (**ML-EM**). We also performed HAC-BIC [4] as the existing method (**BIC**). In the BIC-based method, we carried out the agglomerative merging procedure until the correct number of speakers had been obtained.

The number of iterations was set to 50 for TIMIT and 30 for CSJ.

We considered the first 49 and 29 iterations for TIMIT and CSJ as the burn-in periods, respectively, leading the K values obtained from these periods to be rejected. The K value from the remaining one iteration was then measured. We carried out the same experiment 50 times but using different seeds to generate random numbers and then measured the average of their K values.

Experimental results

Tables 3.2 and 3.3 list the speaker clustering results for TIMIT and CSJ, respectively. The main advantage of Gibbs- and ML-EM-based approaches over BIC is that they assume the existence generative model behind the segments. They therefore can robustly cluster the segments when the number of segments is unbalanced (C-3 and C-4), while BIC based approach easily falls to the local optimum because of the frequent merging between inappropriate pairs of these unbalanced clusters. In addition, Gibbs sampling further improved the speaker clustering performance from ML-EM since the Gibbs sampling could avoid local optimum solutions unlike the ML-EM algorithm.

3.4.3 Speaker clustering experiments in which the number of speakers is unknown

Experimental setup

The hyper-parameters in Eq. 4.6 were set as follows: $h^0 = 1$, $\xi^0 = 1$, and $\eta^0 = 1$. $\boldsymbol{\mu}_i^0$ and $\boldsymbol{\Sigma}_i^0$ were computed as the mean and covariance of all data used in the database. In this experiment, we first estimated the optimal number of clusters as well as the optimal assignments of segments to clusters. Next, we carried out the speaker clustering experiments using the TIMIT and CSJ databases. We then cross-validated for each pair of {T-1, T-2}, {C-1, C-2}, and {C-3, C-4} to decide the penalty parameter in the BIC-based method and the hyper-parameter h^0 in SO-DPMM.

Table 3.4: Speaker clustering results for TIMIT. #cl. denotes the number of clusters estimated.

Eval.	Method	#cl.	ACP	ASP	K
T-1	SO-DPMM	32.4	0.84	0.72	**0.78**
(spkr:24, utt:192)	HAC-BIC	34.0	0.85	0.71	**0.78**
T-2	SO-DPMM	145.0	0.53	0.55	**0.54**
(spkr:144, utt:1,152)	HAC-BIC	174.0	0.54	0.49	0.52

Table 3.5: Speaker clustering results for CSJ. #cl. denotes the number of clusters estimated.

Eval.	Method	#cl	ACP	ASP	K	DER [%]
C-1	SO-DPMM	9.15	0.96	0.78	**0.87**	**0.13**
(spkr:5, utt:500)	HAC-BIC	9.50	0.85	0.72	0.78	0.25
C-2	SO-DPMM	10.4	0.87	0.84	**0.81**	**0.20**
(spkr:10, utt:1,000)	HAC-BIC	16.5	0.73	0.68	0.70	0.36
C-3	SO-DPMM	10.9	0.91	0.70	**0.80**	**0.23**
(spkr:5, utt:9,333)	HAC-BIC	2.00	0.21	0.55	0.34	0.72
C-4	SO-DPMM	13.7	0.73	0.68	**0.71**	**0.28**
(spkr:10, utt:15,435)	HAC-BIC	4.00	0.12	0.29	0.19	0.83

Experimental results

Table 3.4 lists the speaker clustering results for TIMIT. These results show that SO-DPMM outperformed BIC-HAC in terms of estimating the number of speakers for both T-1 and T-2. SO-DPMM also outperformed BIC-HAC in terms of the K value for T-2. Table 3.5 shows the speaker clustering results for CSJ. SO-DPMM outperformed BIC-HAC for all evaluation sets. Specifically, BIC-HAC performed considerably worse for C-3 and C-4. These results indicate that SO-DPMM can be robustly estimated for the unbalanced and large-scale data while BIC-HAC significantly diminishes the clustering accuracy for these data.

Next, we discuss the convergence of the sampling procedure in SO-DPMM. For that purpose, experiments were conducted with the same dataset but different seeds of a pseudo-random number generator. It is of interest whether all trials obtained by the proposed sampling procedure converge to the unique solution. In order to evaluate this convergence, we applied proposed approach to the same dataset with different

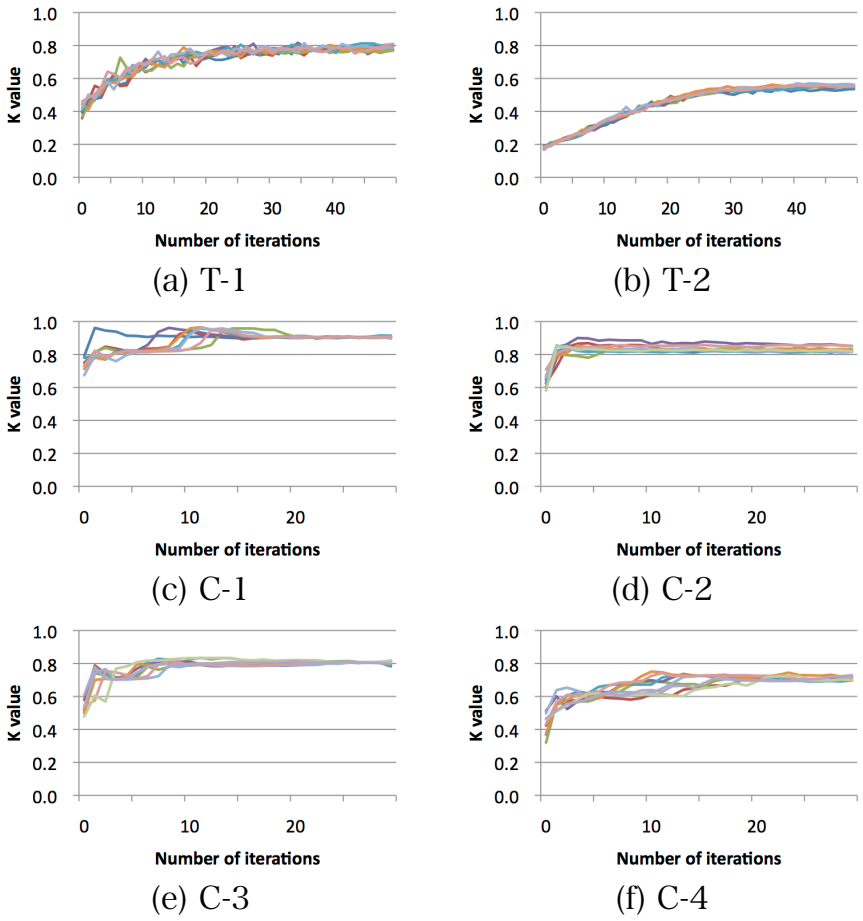


Figure 3.2: K values obtained from proposed method for (a) T-1, (b) T-2, (c) C-1, (d) C-2, (e) C-3 and (f) C-4. Eight lines in each figure show results of eight trials using different seeds.

seeds of pseudo-random number generator. Figure 3.2 shows the K values obtained from SO-DPMM. The eight lines in each figure show the respective results from the eight trials using the different seeds. This figure shows that all samples from all trials converge to the unique distributions. This result indicates that the proposed method is robust against the local optima problem depending on the initial states.

Finally, we discuss computational costs. In the experiment for C-4, SO-DPMM took 11.8 seconds per iteration and 588 seconds for 50 iterations on average when Intel Xeon 3.00 GHz was used. SO-DPMM required comparatively less computation time because of its fast convergence, although sampling-based methods generally require many iterations until the value of the samples converges. Figure 3.2 shows

that all samples converge to the unique distributions within 30 iterations for all datasets. The high effectiveness of the proposed sampling method is attributed to segment-oriented sampling. The slow convergence speed in the general Gibbs sampler is attributed to its sampling procedure in which only one sample is reassigned in each iteration. In the segment-oriented sampling, by contrast, a set of frames is simultaneously reassigned in each iteration. The advantage of SO-DPMM is yielded by using segment-oriented sampling. The general Gibbs sampler induces the slow convergence speed due to its sampling procedure in which only one sample is reassigned in each iteration. In contrast, the segment-oriented sampling simultaneously reassigns a set of frames in each iteration.

3.5 Summary

In this chapter, we formalized the segment-generative model as a mixture of Gaussian distributions by modeling each cluster as a single Gaussian distribution. We, then, demonstrated that segment-generative model can be easily extended to a nonparametric Bayesian model called segment-oriented DPMM (SO-DPMM) by taking the limit of clusters. The optimal structure of SO-DPMM was estimated effectively by using the MCMC-based method and the number of speaker clusters was automatically determined according to the data. In speaker clustering when the number of speakers is unknown, the proposed method was shown to outperform the conventional method, especially for unbalanced and large-scale data.

Chapter 4

Segment-oriented generative clustering via a mixture distribution

4.1 Introduction

In the previous chapter, each cluster is modeled as a Gaussian distribution. However, the underlying assumption of unimodality for this distribution is sometimes too restrictive. For example, short time fast Fourier transforms of acoustic signals, and filter responses in images are known to follow multi-modal distributions, which cannot be represented by unimodal distributions [38, 39, 40]. Mixture models are reasonable approximations for representing these multi-modal distributions [18, 19] and various distributions have been used as components of mixture models such as the t-distribution [20] and von Mises-Fisher distribution [21, 22]. In particular, Gaussian distributions are used widely as a reasonable approximations for a wide class of probability distributions [23]. By using a Gaussian mixture model (GMMs) to represent each cluster, the whole speaker space is modeled as a mixture of these GMMs. We refer to this as a *mixture-of-GMMs (MoGMMs)* .

In this chapter, we extend the previous finite-speaker model to the MoGMMs. In this model, the optimal assignment of segments to clusters can be obtained by evaluating the posterior probability of assigning each observation to each cluster's mixture distribution. Thus, the clustering problem is reduced to the problem of estimating this MoGMMs. It should be noted that mixture-of-mixture models can represent this multi-level property within their hierarchical structure, since each segment-wise observation (e.g., a segment) is assigned to one of the mixture distributions

while each frame-wise observation in a segment is assigned to one of the mixture components in the mixture distribution. The main advantage of introducing mixture distributions for cluster representation is that we can model inner-cluster variability in each cluster by the cluster specific mixture distribution.

To estimate the structure of MoGMMs, frame-level latent variables (fLVs) V is introduced to represent the assignment of frame to cluster's component (i.e. Gaussian distribution in the GMM), as well as the sentence-level latent variables (sLVs) Z that represents the assignment of segment to clusters (i.e. GMM). Both MCMC- and VB-based approaches are based on the estimation of the posterior distribution of latent variables. This approach proposed a MCMC-based model estimation method for MoGMMs that stochastically estimate the posterior distribution by iteratively drawing their samples. We showed that this approach can efficiently estimate the parameters of MoGMMs avoiding local optimum. This sampling-based procedure makes models more robust against the data bias because a large number of trials are evaluated over the obtained samples. However, in practical implementations of the MCMC, evaluating such a huge number of combinations of sLVs and fLVS is infeasible and some approximations are required [41, 34, 42].

Collapsed Gibbs sampling introduced in the previous section is the simplest way to implement MCMC for obtaining sLVs and fLVs from their joint posterior distributions. Actually, Watanabe *et. al* introduced a Gibbs sampling-based MoGMMs estimation approach, which draws the values of fLVs and sLVs alternately by first sampling the fLVs after initializing sLVs Z, i.e., $V \sim p(V|Z)$, and then sampling sLVs by using the fixed fLVs, i.e., $Z \sim p(Z|V)$, sampled in the previous step [31]. However, this sampling method, has a severe restriction because the sampling of sLVs is strictly determined by the values of the fLVs obtained in the previous sampling step. This restriction can cause the local optima problem for the sLVs, because the sLVs estimated in each iteration can be highly correlated. If we relax this constraint, we then make it possible to efficiently evaluate these samples. To solve this problem, this chapter proposes a

novel sampling method for the MoGMMs based on nested Gibbs sampling, which samples both the sLVs and fLVs at the same time. This sampling method allow an enormous number of combinations of fLVs and sLVs to be evaluated efficiently, so a more appropriate solution can be obtained than that obtained by alternating Gibbs sampling for fLVs and sLVs.

The reminder of this chapter is organized as follows. Section 4.2 formulates a MoGMMs by creating a mixture-of-mixture model where each component of the mixtures is represented by a GMM. Section 4.3 explains how to estimate the MoGMMs using fully Bayesian approaches based on VB and MCMC methods. Section 4.4 describes the MCMC-based model estimation method in more detail as well as the proposed nested Gibbs sampling method for MoGMMs estimation. Section 4.5 presents the results of speaker clustering experiments conducted to demonstrate the effectiveness of the proposed method. Section 4.6 summarizes this chapter and discuss some directions for future research.

4.2 Formulation of MoGMMs

This section defines the MoGMMs models where each component of the model is represented by a GMM. In addition, the generative model for segment-oriented data is also defined In this generative model, each segment-wise observation is stochastically assigned to one of the GMM, assigning the frame-wise observations to each cluster's components. Assignment of segment to GMM cluster is assumed as speaker clustering.

4.2.1 Mixture of Gaussian mixture models (MoGMMs)

Let $\mathbf{o}_{ut} \in \mathbb{R}^D$ be a D-dimensional observation vector at the t-th frame in the u-th segment, $\mathbf{O}_u \triangleq \{\mathbf{o}_{ut}\}_{t=1}^{T_u}$ is the u-th segment comprising the T_u observation vectors, and $\mathcal{O} \triangleq \{\mathbf{O}_u\}_{u=1}^{U}$ is a set of U segments. Here, a MoGMMs is defined as follows:

$$p(\mathcal{O}|\mathbf{\Theta}) = \prod_{u=1}^{U} \sum_{i=1}^{S} h_i p(\mathbf{O}_u|\mathbf{\Theta}_i), \tag{4.1}$$

where S denotes the number of clusters; h_i represents how frequently the i-th cluster's segment appears; and $p(\mathbf{O}_u|\mathbf{\Theta}_i)$ is the likelihood of u-th segment $\mathbf{O}_u$ being assigned to the i-th cluster. In this case, $p(\mathbf{O}_u|\mathbf{\Theta}_i)$ models the intra-cluster variability for each cluster, which can be represented as:

$$p(\mathbf{O}_u|\mathbf{\Theta}_i) = \prod_{t=1}^{T_u} \sum_{j=1}^{K} w_{ij} \mathcal{N}(\mathbf{o}_{ut}|\boldsymbol{\mu}_{ij}, \mathbf{\Sigma}_{ij}), \tag{4.2}$$

where $\mathcal{N}$ denotes the j-th component in the i-th cluster, which is represented by a Gaussian distribution with a mean vector $\boldsymbol{\mu}_{ij}$ and a covariance matrix $\mathbf{\Sigma}_{ij}$; w_{ij}, the weight of the j-th component; and K is the number of components in each cluster's GMM. Eqs. 4.1 and 4.2 imply that the whole generative model for all segments $\mathcal{O}$ can be represented by a hierarchically structured MoGMMs where a GMM represents a cluster's characteristics (i.e., intra-cluster variability), and that a mixture of these GMMs can represent the entire cluster space (i.e., inter-cluster variability). In this model, the number of mixtures in the MoGMMs indicates the number of speakers.

To represent this hierarchical model, two types of latent variables are introduced: $\mathcal{Z} = \{z_u\}_{u=1}^{U}$ represents segment-level latent variables (sLVs), each of which identifies a MoGMMs component (i.e., speaker GMM) to which the u-th segment is assigned, and $\mathcal{V} = \{\mathcal{V}_u = \{v_{ut}\}_{t=1}^{T_u}\}_{u=1}^{U}$, represents the frame-level latent variables (fLVs), each of which identifies an intra-cluster GMM component (the cluster distribution to which the u-th segment is assigned), to which the t-th frame-wise observation in the u-th segment is assigned. For instance, the sLVs and fLVs in MoGMMs correspond to the document-level and word-level latent variables in the latent Dirichlet allocation (LDA), where discrete data are used [43]. By contrast, this research focus on modeling a continuous data space with a MoGMMs in this study.

By introducing these latent variables, the conditional distributions of the observed segments given the latent variables is described as follows

$$p(\mathcal{O}|\mathcal{Z}, \mathcal{V}, \mathbf{\Theta}) = \prod_{u=1}^{U} h_{z_u} \prod_{t=1}^{T_u} w_{z_u v_{ut}} \mathcal{N}(\mathbf{o}_{ut}|\boldsymbol{\mu}_{z_u v_{ut}}, \mathbf{\Sigma}_{z_u v_{ut}}), \tag{4.3}$$

where $\mathbf{\Theta} \triangleq \left\{\{h_i\}, \{w_{ij}\}, \left\{\boldsymbol{\mu}_{ij}\right\}, \{\mathbf{\Sigma}_{ij}\}\right\}$ denote the weight of the i-th intra-cluster GMM, weight, mean vector, and covariance matrix of the j-th component of the i-th intra-cluster GMM, respectively. Note that we have assumed $\mathbf{\Sigma}_{ij}$ is a diagonal covariance matrix where the (d,d)-th element is represented by $\sigma_{ij,d}$.

We describe the distribution of the latent variables as follows:

$$P(\mathcal{V}|\mathcal{Z},\mathbf{w}) = \prod_{u=1}^{U}\prod_{t=1}^{T_u}\prod_{i=1}^{S}\prod_{j=1}^{K} w_{ij}^{\delta(v_{ut},j)\delta(z_u,i)}, \tag{4.4}$$

$$P(\mathcal{Z}|\mathbf{h}) = \prod_{u=1}^{U}\prod_{i=1}^{S} h_i^{\delta(z_u,i)}, \tag{4.5}$$

where $\delta(a,b)$ denotes Kronecker's delta, which takes a value of one if $a=b$, and zero otherwise.

4.2.2 Generative process and graphical model

Using a Bayesian approach, the conjugate prior distributions of the parameters are introduced as follows:

$$p(\mathbf{\Theta}|\mathbf{\Theta}^0) = \begin{cases} \mathbf{h} & \sim \mathcal{D}(\mathbf{h}^0) \\ \mathbf{w}_i & \sim \mathcal{D}(\mathbf{w}^0) \\ \{\mu_{ij,d}, \Sigma_{ij,d}\} & \sim \mathcal{NG}(\xi^0, \eta^0, \mu_{j,d}^0, \sigma_{j,d}^0) \end{cases} \tag{4.6}$$

where $\mathcal{D}\left(\mathbf{h}^0\right)$ and $\mathcal{D}\left(\mathbf{w}^0\right)$ denote Dirichlet distributions with hyper-parameters $\mathbf{h}^0$ and $\mathbf{w}^0$, respectively. $\mathcal{NG}\left(\xi^0, \eta^0, \mu_{j,d}^0, \sigma_{j,d}^0\right)$ denotes the normal inverse gamma distribution with hyper-parameters ξ^0, η^0, $\mu_{j,d}^0$, and $\sigma_{j,d}^0$ [1].

Based on these likelihoods and prior distributions, the generative process for our model is described as follows:

1. Initialize $\left\{\mathbf{h}^0, \mathbf{\Theta}^0\right\}$
2. Draw $\mathbf{h}$ from $\mathcal{D}(\mathbf{h}^0)$
3. For each segment-level mixture component (i.e., cluster) $i = 1, \cdots, S$,
   (a) Draw $\mathbf{w}_i$ from $\mathcal{D}(\mathbf{w}^0)$
   (b) For each frame-level mixture component (i.e., inner-cluster GMM component) $j = 1, \cdots, K$,

[1] The detailed definition of Dirichlet and Gaussian-Gamma distributions is described in Appendix

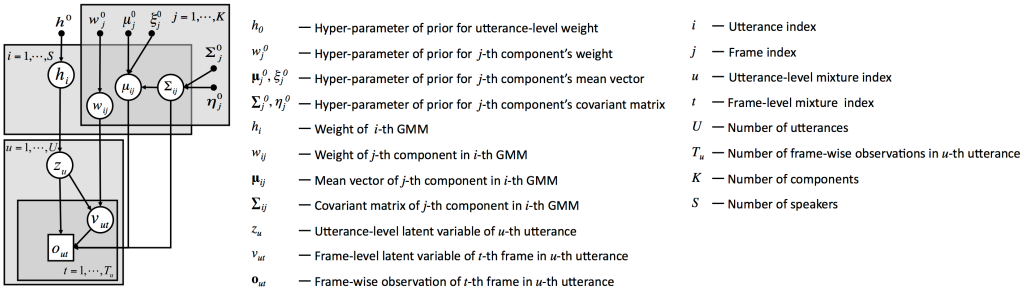


Figure 4.1: *Graphical representation of mixture-of-mixture model. The white square denotes frame-wise observations, and dots denote the hyper-parameters of prior distributions.*

i. Draw $\{\mu_{i,d}, \sigma_{i,d}\}$ from $\mathcal{N}G(\xi_j^0, \eta_j^0, \mu_{j,d}^0, \sigma_{j,d}^0)$ for each dimension $d = 1, \cdots, D$

4. For each segment $u = 1, \cdots, U$,

   (a) Draw z_u from multinomial distribution $\mathcal{M}(\mathbf{h})$

   (b) For each frame $t = 1, \cdots, T_u$,

      i. Draw v_{ut} from $\mathcal{M}(\mathbf{w}_{z_u})$

      ii. Draw $\mathbf{o}_{ut}$ from $\mathcal{N}(\boldsymbol{\mu}_{z_u v_{ut}}, \boldsymbol{\Sigma}_{z_u v_{ut}})$

Figure 4.1 shows a graphical representation of this model.

4.3 Model inference based on fully Bayesian approach

When we use a Bayesian approach for estimating the MoGMMs, the main task is calculating posterior distributions for the latent variables $\{\mathcal{V}, \mathcal{Z}\}$ and model parameter $\boldsymbol{\Theta}$ given observation $\mathcal{O}$:

$$p(\mathcal{V}, \mathcal{Z}, \boldsymbol{\Theta}|\mathcal{O}) = \frac{1}{H_0} p(\mathcal{O}, \mathcal{V}, \mathcal{Z}, \boldsymbol{\Theta}). \tag{4.7}$$

H_0 is a normalization coefficient, which is defined as follows:

$$H_0 \triangleq p(\mathcal{O}) = \sum_{\mathcal{V}, \mathcal{Z}} \int p(\mathcal{O}, \mathcal{V}, \mathcal{Z}, \boldsymbol{\Theta}) \mathrm{d}\boldsymbol{\Theta}. \tag{4.8}$$

Note that the model-based clustering problem is reduced to the problem of estimating the optimal values of the fLVs and sLVs, $\{\mathcal{V}, \mathcal{Z}\}$, based

on the posterior distribution defined in Eq. 4.7. Thus, the posterior probabilities of the latent variables $\mathcal{V}$ and $\mathcal{Z}$ can be calculated as follows:

$$\gamma_{v_{ut}=j|z_u=i;\boldsymbol{\Theta}} \triangleq p(v_{ut}=j|\mathbf{O},\boldsymbol{\Theta},z_u=i), \tag{4.9}$$

$$\gamma_{z_u=i;\boldsymbol{\Theta}} \triangleq p(z_u=i|\mathbf{O},\boldsymbol{\Theta}). \tag{4.10}$$

Sufficient statistics of this model are computed using the aforementioned posterior probabilities as follows:

$$\begin{cases} c_i & = \sum_u \gamma_{z_u=i} \\ n_{ij} & = \sum_{u,t} \gamma_{v_{ut}=j|z_u=i} \cdot \gamma_{z_u=i} \\ \mathbf{m}_{ij} & = \sum_{u,t} \gamma_{v_{ut}=j|z_u=i} \cdot \gamma_{z_u=i} \cdot \mathbf{o}_{ut} \\ r_{ij,d} & = \sum_{u,t} \gamma_{v_{ut}=j|z_u=i} \cdot \gamma_{z_u=i} \cdot (o_{ut,d})^2 \end{cases} \tag{4.11}$$

where c_i denotes the number of segments assigned to the i-th component of the entire cluster MoGMMs; n_{ij} is the number of frames assigned to the j-th component of the i-th intra-cluster GMM of the MoGMMs; and m_{ij} and r_{ij} are the first and second sufficient statistics, respectively.

However, it is generally infeasible to analytically estimate these posterior distributions, so we must introduce some approximations. In the rest of this section, we discuss how to approximate the posterior distributions using VB- and MCMC-based approaches.

4.3.1 Model estimation using a VB-based approach

When the VB-based model estimation method is used, the sLVs, fLVs, and model parameters are obtained deterministically by estimating their variational posterior distributions. To optimize a variational posterior distribution, we attempt to maximize the marginalized likelihood, which is described by Eq. 4.8. The lower bound of the marginalized logarithmic likelihood (i.e., $\log p(\mathcal{O})$) is obtained as follows:

$$\mathcal{F}[q(\mathcal{V},\mathcal{Z},\boldsymbol{\Theta})] = \left\langle \log \frac{p(\mathcal{V},\mathcal{Z},\boldsymbol{\Theta}|\mathcal{O})}{q(\mathcal{V},\mathcal{Z},\boldsymbol{\Theta}|\mathcal{O})} \right\rangle_{q(\mathcal{V},\mathcal{Z},\boldsymbol{\Theta}|\mathcal{O})} \tag{4.12}$$

Under the assumption that each variable in the variational posterior distribution is independent and identically distributed,

$$p(\mathcal{V},\mathcal{Z},\boldsymbol{\Theta}|\mathcal{O}) = q(\mathcal{Z})q(\mathcal{V}|\mathcal{Z})q(\boldsymbol{\Theta})q(\mathcal{V},\mathcal{Z})q(\boldsymbol{\Theta}), \tag{4.13}$$

where the optimal variational posterior distribution (i.e., the $q(\mathcal{V}, \mathcal{Z}, \mathbf{\Theta})$ that maximizes the free energy) can be determined as follows:

$$q(\mathcal{V}|\mathcal{Z}) \propto \exp\left(\left\langle \log p(\mathcal{O}, \mathcal{V}, \mathcal{Z}, \mathbf{\Theta})\right\rangle_{q(\mathbf{\Theta})}\right), \tag{4.14}$$

$$q(\mathcal{Z}) \propto \exp\left(\left\langle \log p(\mathcal{O}, \mathcal{V}, \mathcal{Z}, \mathbf{\Theta})\right\rangle_{q\mathcal{V})q(\mathbf{\Theta})}\right), \tag{4.15}$$

$$q(\mathbf{\Theta}) \propto \exp\left(\left\langle \log p(\mathcal{O}, \mathcal{V}, \mathcal{Z}, \mathbf{\Theta})\right\rangle_{q(\mathcal{V},\mathcal{Z})}\right). \tag{4.16}$$

where $\langle A\rangle_B$ denotes the expectation of A with respect to B. The optimal values of $q(\mathcal{V}, \mathcal{Z})$, and $q(\mathbf{\Theta})$ from Eqs. 4.14, 4.15 and 4.16 are obtained according to Algorithm 2, as follows. The posterior probability of an fLV is estimated as follows:

$$\begin{aligned}\gamma^*_{v_{ut}=j|z_u=i;\tilde{\mathbf{\Theta}}} \\ \triangleq \exp\Bigg(&\langle \log w_{ij}\rangle_{q(w_{ij})} + \frac{1}{2}\sum\nolimits_d \langle \log \sigma_{ij,d}\rangle_{q(\sigma_{ij,d})} \\ &-\frac{D}{2}\log 2\pi - \frac{1}{2}\sum\nolimits_d \langle \frac{(o_{ut,d}-\mu_{ij,d})^2}{\sigma_{ij,d}}\rangle_{q(\mu_{ij,d}|\sigma_{ij,d})}\Bigg).\end{aligned} \tag{4.17}$$

We can determine the posterior distribution of an fLV by normalizing Eq. 4.17 as follows:

$$q(v_{ut}=j|z_u=i) = \frac{\gamma^*_{v_{ut}=j|z_u=i;\tilde{\mathbf{\Theta}}}}{\sum_j \gamma^*_{v_{ut}=j|z_u=i;\tilde{\mathbf{\Theta}}}}. \tag{4.18}$$

In the same manner, we can compute an sLV $\gamma_{z_u=i;\tilde{\mathbf{\Theta}}}$ from the posterior probability $\gamma^*_{z_u=i;\tilde{\mathbf{\Theta}}}$ as follows:

$$\gamma^*_{z_u=i;\tilde{\mathbf{\Theta}}} \triangleq \exp\left(\langle \log h_i\rangle_{q(h_i)} + \sum_t \log \sum_j \gamma^*_{v_{ut}=j|z_u=i;\tilde{\mathbf{\Theta}}}\right), \tag{4.19}$$

$$q(z_u=i) = \frac{\gamma^*_{z_u=i;\tilde{\mathbf{\Theta}}}}{\sum_i \gamma^*_{z_u=i;\tilde{\mathbf{\Theta}}}}. \tag{4.20}$$

The expected values of the parameters described in (4.17) to (4.19) are computed as follows:

$$\langle \log h_i\rangle_{q(h_i)} = \psi(\tilde{h}_i) - \psi(\sum\nolimits_i \tilde{h}_i), \tag{4.21}$$

$$\langle \log w_{ij}\rangle_{q(w_{ij})} = \psi(\tilde{w}_{ij}) - \psi(\sum\nolimits_j \tilde{w}_{ij}), \tag{4.22}$$

$$\langle \log \sigma_{ij,d}\rangle_{q(\sigma_{ij,d})} = \psi(\tilde{\eta_{ij}}) - \log \tilde{\sigma}_{ij,d}, \tag{4.23}$$

$$\langle \frac{(o_{ut,d}-\mu_{ij,d})^2}{\sigma_{ij,d}}\rangle_{q(\mu_{ij,d}|\sigma_{ij,d})q(\sigma_{ij,d})} = \frac{\tilde{\eta}_{ij}(o_{ut,d}-\tilde{\mu}_{ij,d})^2 + \tilde{\xi}_{ij}}{\tilde{\sigma}_{ij,d}}, \tag{4.24}$$

Algorithm 2: Model estimation algorithm using the VB method.

```
initialize Θ̃;
repeat
    for all clusters i and components j do
        for all segments u and frames t do
            Compute γ_q(V, Z) in Eq. 4.14 before computing the
            expectation values described in Eqs. 4.18 and 4.20;
        end
    end
    for all clusters i and components j do
        Compute the hyper-parameters of q( ) in Eq. 4.16 using the
        sufficient statistics, as described in Eqs. 4.21 - 4.24;
    end
until converged;
```

where $\psi(\cdot)$ denotes the digamma function and $\tilde{\boldsymbol{\Theta}} = \left\{\tilde{h}_i, \tilde{w}_{ij}, \tilde{\xi}_{ij}, \tilde{\eta}_{ij}, \tilde{\boldsymbol{\mu}}_{ij}\right\}$ are the hyper-parameters of the posterior distributions for $\tilde{\boldsymbol{\Theta}}$, which are computed as follows:

$$\tilde{\boldsymbol{\Theta}} = \begin{cases} \tilde{h}_i & = h^0 + c_i, \\ \tilde{w}_{ij} & = w_j^0 + n_{ij}, \\ \tilde{\xi}_{ij} & = \xi^0 + n_{ij}, \\ \tilde{\eta}_{ij} & = \eta^0 + n_{ij}, \\ \tilde{\boldsymbol{\mu}}_{ij} & = \tilde{\xi}_{ij}^{-1}\left(\xi^0 \boldsymbol{\mu}_j^0 + \mathbf{m}_{ij}\right), \\ \tilde{\sigma}_{ij,d} & = \sigma_{j,d}^0 + r_{ij,d} + \xi^0 (\mu_{j,d}^0)^2 - \tilde{\xi}_{ij} (\tilde{\mu}_{ij,d})^2 \end{cases} \tag{4.25}$$

Algorithm 2 shows the VB-based model estimation algorithm. The fLVs and sLVs that maximize (Eqs. 4.18 and 4.20) are the MAP values of their posterior distributions, where we assume that these MAP values are the optimal clustering results. This variational calculation is shown in detail in Appendix B.

4.3.2 Model estimation based on the MCMC approach

Using an MCMC-based approach, we obtain samples of latent variables directly from their posterior distributions.

Marginalized likelihood for complete data

First, we derive the logarithmic marginalized likelihood for the complete data, $\log p(\mathcal{O}, \mathcal{V}, \mathcal{Z})$. In the case of complete data, we can utilize

all the alignments of observations $\mathbf{o}_{ut}$ to a specific Gaussian component distribution because all of the latent variables, $\{\mathcal{V}, \mathcal{Z}\}$, are treated as observations. Then, the posterior distributions for each of the latent variables, $P(z_u = i|\cdot)$ and $P(v_{ut} = j|\cdot)$ for all i, j, u, and t, return 0 or 1 based on the assigned information. Thus, $\gamma_{v_{ut}=j|z_u=i}$ and $\gamma_{z_u=i}$ described by Eqs. 4.9 and 4.10 are zero-or-one values depending on the assignment of the data. Then, the sufficient statistics of this model, then, can be represented as follows:

$$\begin{cases} c_i & = \sum_u \delta(z_u, i), \\ n_{ij} & = \sum_{u,t} \delta(z_u, i) \cdot \delta(v_{ut}, j), \\ \mathbf{m}_{ij} & = \sum_{u,t} \delta(z_u, i) \cdot \delta(v_{ut}, j) \cdot \mathbf{o}_{ut}, \\ r_{ij,d} & = \sum_{u,t} \delta(z_u, i) \cdot \delta(v_{ut}, j) \cdot (o_{ut,d})^2, \end{cases} \tag{4.26}$$

We can analytically derive the logarithmic marginalized likelihood for the complete data by substituting Eqs. 4.3, 4.4, and 4.5 into the following integration equation:

$$\begin{aligned} \log p(\mathcal{V}, \mathcal{Z}, \mathcal{O}) &= \log \int p(\mathcal{V}, \mathcal{Z}, \mathcal{O}|\mathbf{\Theta}) p(\mathbf{\Theta}) \mathrm{d}\mathbf{\Theta} \\ &= \log \frac{\Gamma(h^0) \prod_i \Gamma(\tilde{h}_i)}{\Gamma(h^0)^S \Gamma(\sum_i \tilde{h}_i)} + \log \prod_i \frac{\Gamma(\sum_j w_j^0) \prod_j \Gamma(\tilde{w}_{ij})}{\prod_j \Gamma(w_j^0) \Gamma(\sum_j \tilde{w}_{ij})} \\ &\quad + \beta \log \prod_{i,j} (2\pi)^{-\frac{n_{ij} D}{2}} \frac{(\xi^0)^{\frac{D}{2}} \left(\Gamma\left(\frac{\eta_j^0}{2}\right)\right)^{-D} \left(\prod_d \sigma_{j,dd}^0\right)^{\frac{\eta_j^0}{2}}}{(\tilde{\xi}_{ij})^{\frac{D}{2}} \left(\Gamma\left(\frac{\tilde{\eta}_{ij}}{2}\right)\right)^{-D} \left(\prod_d \tilde{\sigma}_{ij,dd}\right)^{\frac{\tilde{\eta}_{ij}}{2}}}, \end{aligned} \tag{4.27}$$

where $\tilde{\mathbf{\Theta}}_{ij} \triangleq \left\{\tilde{h}_i, \tilde{w}_{ij}, \tilde{\xi}_{ij}, \tilde{\eta}_{ij}, \tilde{\mu}_{ij,d}, \tilde{\sigma}_{ij,d}\right\}$ denotes the hyper-parameter of the marginalized likelihood defined in Eq. 4.25.

To construct the MCMC sampler, we define the following logarithmic likelihood function for the complete data using simulated annealing (SA) [44]:

$$\begin{aligned} H_p(\beta) &\triangleq \log p_\beta(\mathcal{V}, \mathcal{Z}, \mathcal{O}) \\ &= \log p(\mathcal{O}|\mathcal{V}, \mathcal{Z}) + \frac{1}{\beta} \log P(\mathcal{V}, \mathcal{Z}), \end{aligned} \tag{4.28}$$

where β is an inverse temperature defined for SA, which controls the speed of convergence. We can now derive the posterior distribution as

follows:

$$\begin{aligned} P(\mathcal{V}, \mathcal{Z}|\mathcal{O}) &= \frac{1}{H_p(\beta)} p(\mathcal{V}, \mathcal{Z}) p(\mathcal{O}|\mathcal{V}, \mathcal{Z})^{\beta} \\ &= \frac{1}{H_p(\beta)} \exp\left\{-\beta H(\mathbf{\Psi})\right\}, \end{aligned} \tag{4.29}$$

where $H_p(\beta)$ is a normalization term introduced to normalize $\{\mathcal{V}, \mathcal{Z}\}$ under the temperature β. The goal of the MCMC approach is to obtain samples from Eq. 4.29. In the next section, we discuss how to design the sampler in order to obtain samples from this posterior distribution.

4.4 Implementation of MCMC-based model estimation

We introduce a collapsed Gibbs sampler [34] to obtain samples of sLVs and fLVs from their posterior distributions. One simple approach is to introduce a Gibbs assumption that alternates the sampling of fLVs with some initializations of sLVs, before sampling the sLVs using the fixed fLVs sampled in the previous step as proposed in [31]. The drawback of this approach is that the sampling of sLVs is determined strictly by the values of the fLVs obtained in the previous sampling step and the sLVs estimated in each iteration can be highly correlated. To solve this problem, we propose a novel sampling method that samples both sLVs and fLVs at the same time. This sampling method allows an enormous number of combinations of fLVs and sLVs to be evaluated efficiently, so we can find a more appropriate solution than that obtained when alternating Gibbs sampling for fLVs and sLVs. We refer to this novel sampling technique as nested Gibbs sampling. This section describes its formulation and implementation.

4.4.1 Nested Gibbs sampling for MoGMMs

For Gibbs sampling, we draw the value of each variable iteratively from its posterior distributions and conditioning it with the sampled values of the other variables. In the case of MoGMMs, fLVs and sLVs are unknown. Therefore, the proposal distribution is the joint posterior distribution of

the sLV and fLVs related to the u-th segment of $\{\mathcal{V}_u, z_u\}$, which is conditioned on the sampled value of the latent variables related to the other segments $\left\{\mathcal{V}^*_{\backslash u}, \mathcal{Z}^*_{\backslash u}\right\}$. Therefore, the proposal distribution of MoGMMs is described as follows:

$$P\left(\mathcal{V}_u, z_u | \left\{\mathcal{V}^*_{\backslash u}, \mathcal{Z}^*_{\backslash u}\right\}, \mathcal{O}\right) = \frac{p\left(\mathcal{V}_u, z_u, \left\{\mathcal{V}^*_{\backslash u}, \mathcal{Z}^*_{\backslash u}\right\}, \mathcal{O}\right)}{p\left(\left\{\mathcal{V}^*_{\backslash u}, \mathcal{Z}^*_{\backslash u}\right\}, \mathcal{O}\right)}, \tag{4.30}$$

where $\mathcal{V}^*_{\backslash u} = \{v^*_{u't} | \forall u' \neq u, \forall t\}$ and $\mathcal{Z}^*_{\backslash u} = \{z^*_{u'} | \forall u' \neq u\}$ denote the sets of samples for fLVs and sLVs, respectively, except for those related to the u-th segment. After some iterative sampling using Eq. 4.30, the samples obtained are approximately distributed according to their true posterior distributions. Direct sampling from a proposal distribution Eq. 4.30 is theoretically feasible because Eq. 4.30 takes the form of a multinomial distribution, and we can evaluate the value of Eq. 4.30 using Eq. 4.27 for all possible combinations of $\{\mathcal{V}_u, z_u\}$). However, it is impractical to evaluate an enormous number of possible combinations of solutions.

We notice that it is enough to estimate the value of sLVs in order to estimate the optimal assignment of segments to speaker clusters. Therefore, we try to marginalize out fLVs in Eq. 4.30 to make the computation simple. We propose an MCMC-based approach, which samples the value of z_u directly from the following marginalized posterior instead of Eq. 4.30:

$$\begin{aligned} & p\left(z_u | \left\{\mathcal{V}^*_{\backslash u}, \mathcal{Z}^*_{\backslash u}\right\}, \mathcal{O}\right) \\ & = \int p\left(z_u | \mathcal{V}^*_u, \left\{\mathcal{V}^*_{\backslash u}, \mathcal{Z}^*_{\backslash u}\right\}, \mathcal{O}\right) p\left(\mathcal{V}^*_u | \left\{\mathcal{V}^*_{\backslash u}, \mathcal{Z}^*_{\backslash u}\right\}, \mathcal{O}\right) \mathrm{d}\mathcal{V}_u \end{aligned} \tag{4.31}$$

However, this integration is also infeasible because each v_{ut} in $\mathcal{V}_u$ takes one of the number of K values (i.e., the number of GMM components) and their combination are exponentially large. Therefore, we introduce an approximated approach, which uses the sampled value of $\mathcal{V}^{**}_u$ obtained from its true posterior $p\left(\mathcal{V}^*_u | \left\{\mathcal{V}^*_{\backslash u}, \mathcal{Z}^*_{\backslash u}\right\}, \mathcal{O}\right)$. Then, $\mathcal{V}^*_u$ is marginalized out from Eq. 4.30 using the sampled value $\mathcal{V}^{**}_u$ by the following approximation:

$$p\left(z_u | \left\{\mathcal{V}^*_{\backslash u}, \mathcal{Z}^*_{\backslash u}\right\}, \mathcal{O}\right) \simeq \sum_{\mathcal{V}^{**}_u} P\left(z_u | \mathcal{V}^{**}_u, \left\{\mathcal{V}^*_{\backslash u}, \mathcal{Z}^*_{\backslash u}\right\}, \mathcal{O}\right) \tag{4.32}$$

We can easily sample the value of z_u from Eq. 4.32 because this is a multinomial distribution over z_u that takes the one of the C (i.e., number of clusters) values.

With this approach, the Gibbs sampling chain for z_u is followed by another Gibbs sampling chain in which we sample the values of $\mathcal{V}_u$ from its posterior distribution, conditioned on any potential value of z_u. We refer to this Gibbs sampler for $\mathcal{V}_u$ as a sub-Gibbs sampler and we refer to the obtained samples as $\mathcal{V}^{**}_{u|z_u=i} = \left\{v^{**}_{ut|z_u=i}\right\}_{t=1}^{T_u}$. In the sub-Gibbs sampler, each value of $v^{**}_{ut|z_u=i}$ is sampled for all i as follows:

$$v^{**}_{ut|z_u=i} \sim P\left(v_{ut}|\left\{\mathcal{V}^{*}_{\setminus u}, \mathcal{Z}^{*}_{\setminus u}\right\}, \mathcal{V}^{**}_{u\setminus t}, z_u = i, \mathcal{O}\right), \tag{4.33}$$

where $\mathcal{V}^{**}_{u\setminus t|z_u=i} = \left\{v^{**}_{ut'|z_u=i}|\forall t' \neq t\right\}$ denotes the samples of fLVs obtained from the sub-Gibbs sampler that are related to all of the frames, except to the t-th frame in the u-th segment. After several iterations of Eq. 4.33 for all t in the u-th segment, we obtain N^{Gibbs} samples. We then draw a sample of sLV for the u-th segment from its posterior distribution conditioned on the samples $\left\{\mathcal{V}^{**}_{u|z_u=i}{}^{(n)}\right\}_{n=1}^{N^{Gibbs}}$. By aggregating the value of $\left\{\mathcal{V}^{**}_{u|z_u=i}{}^{(n)}\right\}_{n=1}^{N^{Gibbs}}$ over all possible values of i, the Gibbs sampler for z_u is defined as follows:

$$\begin{aligned} z_u &\sim P(z_u|\left\{\mathcal{V}^{*}_{\setminus u}, \mathcal{Z}^{*}_{\setminus u}\right\}, \mathcal{O}) \\ &= \sum_{\forall \mathcal{V}_u} P\left(\mathcal{V}_u, z_u|\left\{\mathcal{V}^{*}_{\setminus u}, \mathcal{Z}^{*}_{\setminus u}\right\}, \mathcal{O}\right) \\ &= \sum_{\forall \mathcal{V}_u} P\left(z_u|\left\{\mathcal{V}^{*}_{\setminus u}, \mathcal{Z}^{*}_{\setminus u}\right\}, \mathcal{V}_u, \mathcal{O}\right) P\left(\mathcal{V}_u|\left\{\mathcal{V}^{*}_{\setminus u}, \mathcal{Z}^{*}_{\setminus u}\right\}, \mathcal{O}\right) \end{aligned} \tag{4.34}$$

By aggregating N^{Gibbs} samples of $\mathcal{V}_u$ from $p\left(\mathcal{V}_u|z_u = i, \left\{\mathcal{V}^{*}_{\setminus u}, \mathcal{Z}^{*}_{\setminus u}\right\}, \mathcal{O}\right)$ for all possible values of i and then plugging them into $p(z_u|V_u, O)$, we obtain the following Monte Carlo integration:

$$\begin{aligned} &\sum_{\forall \mathcal{V}_u} P\left(z_u|\left\{\mathcal{V}^{*}_{\setminus u}, \mathcal{Z}^{*}_{\setminus u}\right\}, \mathcal{V}_u, \mathcal{O}\right) P\left(\mathcal{V}_u|\left\{\mathcal{V}^{*}_{\setminus u}, \mathcal{Z}^{*}_{\setminus u}\right\}, \mathcal{O}\right) \\ &\qquad \simeq \frac{1}{N^{Gibbs}} \sum_{n=1}^{N^{Gibbs}} P\left(z_u|\mathcal{V}^{**(n)}_{u}, \left\{\mathcal{V}^{*}_{\setminus u}, \mathcal{Z}^{*}_{\setminus u}\right\}, \mathcal{O}\right) \end{aligned} \tag{4.35}$$

We refer to these procedures as *nested Gibbs sampling*, because we sample z_u from Eq. 4.35 using the value of $\mathcal{V}^{**(n)}_{u}$ which can be obtained from the sub-Gibbs sampler defined by Eq. 4.33 in a nested manner. A large

number of samples, N^{Gibbs}, may be required to accurately represent of the marginal value for Eq. 4.35. To evaluate the effect of the number of samples on the overall sampling procedure, we applied the proposed nested Gibbs sampler to practical speech data. Figure 4.2 shows the logarithmic marginalized likelihoods (LMLs) obtained using the proposed nested Gibbs sampling method with different sampling sizes. The eight lines in each figure correspond to the results of eight trials with different random seeds. This figure shows that high accuracy may be achieved with a small number of samples, and that even one sample may be adequate to approximate the marginal value in Eq. 4.35. Algorithm 3 shows the algorithm of the nested Gibbs sampler for MoGMMs.

4.4.2 Posterior probability

Here, we provide detailed descriptions of how to calculate the posterior probabilities for the fLVs and sLVs in Eqs. 4.33 and 4.35, which are required for nested Gibbs sampling.

$$
\begin{aligned}
& p(z_u = i | \mathcal{O}, \mathcal{V}_u, \{\mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u}\}) \\
&= \frac{p(\mathcal{O}, \mathcal{V}, \mathcal{Z}_{\backslash u}, z_u = i)}{p(\mathcal{O}, \mathcal{V}, \mathcal{Z}_{\backslash u})} \\
&\propto \frac{p(\mathcal{O}, \mathcal{V}, \mathcal{Z}_{\backslash u}, z_u = i)}{p(\mathcal{O}_{\backslash u}, \mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u})} \\
&\propto \exp\left\{ \log \frac{\Gamma(\sum_j \tilde{w}_{i\backslash u,j})}{\Gamma(\sum_j \tilde{w}_{i,j})} \beta \sum_j \left(H(\tilde{\mathbf{\Psi}}_{i,j}) - H(\tilde{\mathbf{\Psi}}_{i\backslash u,j}) \right) \right\} \\
&\triangleq \gamma^{\beta}_{z_u = i | \mathcal{V}}
\end{aligned}
\tag{4.36}
$$

To derive the result using Eq. 4.36, we assume that the marginalized likelihood for each complete data $\{\mathbf{o}_{ut}, v_{ut}, z_u\}$ is independent from the others, and use the fact that

$$
\begin{aligned}
p(\mathcal{O}, \mathcal{V}_u, \{\mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u}\}) &= p(\mathcal{O}_{\backslash u}, \mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u}) \sum_{z_u} p(\mathcal{O}_u, \mathcal{V}_u, z_u) \\
&\propto p(\mathcal{O}_{\backslash u}, \mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u})
\end{aligned}
\tag{4.37}
$$

$H(\tilde{}_{i,j})$ in Eq. 4.36 denotes the logarithmic likelihood of the complete data $\{\mathcal{O}, \mathcal{Z}, \mathcal{V}\}$, which is defined as follows:

$$\begin{aligned} H(\tilde{}_{i,j}) &\triangleq \log p(\mathcal{O}, \mathcal{V}_{u\backslash t} \left\{\mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u}\right\}, v_{ut} = j, z_u = i) \\ &\propto \log \Gamma(\tilde{w}_{ij}) - \frac{D}{2} \log \tilde{\xi}_{ij} \\ &\quad + D \log \Gamma\left(\frac{\tilde{\eta}_{ij}}{2}\right) - \frac{\tilde{\eta}_{ij}}{2} \sum\nolimits_d \log \tilde{\sigma}_{ij,d} \end{aligned} \tag{4.38}$$

where $\tilde{h}_i$, $\tilde{w}_{ij}$, $\tilde{\xi}_{ij}$, $\tilde{\eta}_{ij}$, $\tilde{\boldsymbol{\mu}}_{ij}$, and $\tilde{\sigma}_{ij,d}$ denote the hyper-parameters of the marginalized likelihood defined in Eq. 4.25. We can also obtain the samples of fLVs from these factorized distributions as follows:

$$\begin{aligned} & p(v_{ut} = j | \mathcal{O}, \mathcal{V}_{u\backslash t}, \left\{\mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u}\right\}, z_u = i) \\ &= \frac{p(\mathcal{O}, \mathcal{V}_{u\backslash t}, \left\{\mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u}\right\}, v_{ut} = j, z_u = i)}{p(\mathcal{O}, \mathcal{V}_{u\backslash t}, \left\{\mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u}\right\}, z_u = i)} \\ &\propto \frac{p(\mathcal{O}, \mathcal{V}_{u\backslash t}, \left\{\mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u}\right\}, v_{ut} = j, z_u = i)}{p(\mathcal{O}_{\backslash\{ut\}}, \mathcal{V}_{\backslash\{ut\}}, \mathcal{Z}_{\backslash u}, z_u = i)} \\ &\propto \exp\left\{-\beta\left(H(\tilde{\boldsymbol{\Psi}}_{i,j}) - H(\tilde{\boldsymbol{\Psi}}_{i,j\backslash t})\right)\right\} \\ &\triangleq \gamma^{\beta}_{v_{ut}=j|z_u=i} \end{aligned} \tag{4.39}$$

where $H(\tilde{\boldsymbol{\Psi}}_{i\backslash u,j})$ $H(\tilde{\boldsymbol{\Psi}}_{i,j\backslash t})$ in Eqs. 4.36 and 4.39 denote the logarithmic likelihood of complete data with respect to $\left\{\mathcal{O}_{\backslash t}, \mathcal{Z}, \mathcal{V}_{\backslash t}\right\}$ and $\left\{\mathcal{O}_{\backslash u}, \mathcal{Z}_{\backslash u}, \mathcal{V}_{\backslash u}\right\}$, respectively.

To derive the result Eq. 4.39, we assume that the marginalized likelihood for each complete data $\{\mathbf{o}_{ut}, v_{ut}, z_u\}$ is i.i.d. and we use the fact that

$$\begin{aligned} & p(\mathcal{O}, \mathcal{V}_{u\backslash t}, \left\{\mathcal{V}_{\backslash u}, \mathcal{Z}_{\backslash u}\right\}, z_u = i) \\ &= p(\mathcal{O}_{\backslash\{ut\}}, \mathcal{V}_{\backslash\{ut\}}, \mathcal{Z}_{\backslash u}, z_u = i) \sum_{v_{ut}} p(\mathbf{o}_{ut}, v_{ut}, z_u = i) \\ &\propto p(\mathcal{O}_{\backslash\{ut\}}, \mathcal{V}_{\backslash\{ut\}}, \mathcal{Z}_{\backslash u}, z_u = i) \end{aligned} \tag{4.40}$$

In the case where we select z_u, for example, we draw the value of z_u from its conditional posterior distribution $p(z_u | \mathcal{O}, \mathcal{V}, \mathcal{Z}_{\backslash u})$ and set the drawn value to z_u. We iterate this procedure for all variables $\mathcal{Z}$ till they converge and we can obtain the samples from objective posterior distribution $P(\mathcal{Z} | \mathcal{V}, \mathcal{O})$.

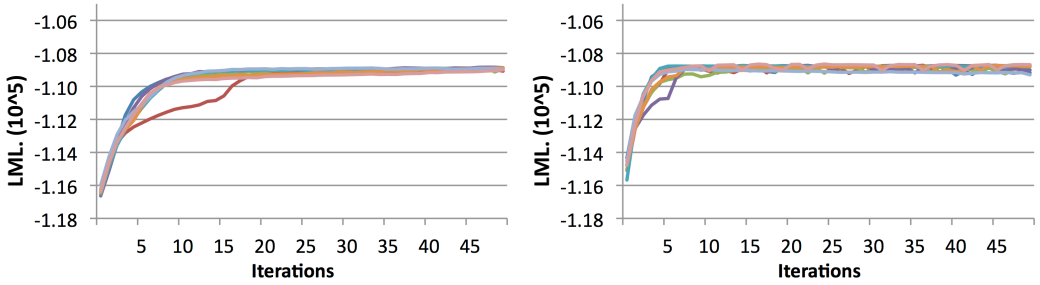
Algorithm 3: Model estimation algorithm based on the proposed nested Gibbs sampling method.

initialization $\{\mathcal{V}^{**}, \mathcal{Z}^{**}\}$, $\mathcal{V}^{*}$;
repeat
  for *all segments* u **do**
    for *all clusters* i **do**
      for *all frames* t **do**
        for *all components* j **do**
          Update
          $\gamma^{\beta}_{v_{ut}=j|z_u=i} \leftarrow P_\beta\left(v_{ut}=j|\left\{\mathcal{Z}^{*}_{\backslash u}, \mathcal{V}^{*}_{\backslash u}\right\}\mathcal{V}^{**}_{u\backslash t}; z_u=i\right)$;
        end
        Draw the values of the fLVs, v^{**}_{ut}, from their posterior probability with $v^{**}_{ut} \sim \gamma^{\beta}_{v_{ut}=\cdot|z_u=i}$;
      end
      Update $\gamma^{\beta}_{z_u=i|\mathcal{V}^{*}_{u}} \leftarrow P_\beta\left(z_u=i|\left\{\mathcal{Z}^{*}_{\backslash u}, \mathcal{V}^{*}_{\backslash u}\right\}, \mathcal{V}^{**}_{u}\right)$;
    end
    Draw the value of the sLVs, z^{*}_{u}, from their posterior distribution with $z^{*}_{u} \sim \gamma^{\beta}_{z_u=i|\mathcal{V}^{**}_{u}}$;
    Update the values of the fLVs with $\mathcal{V}^{*}_{u} \leftarrow \mathcal{V}^{**}_{u}$;
    Update the SA temperature β with respect to scheduling;
  end
until *some conditions are met*;

4.4.3 Computation of the marginalized likelihood

For the Gibbs sampler described in 4.4.1, we can approximate the joint likelihood Eq. 4.28 using the sampled latent variables $\{\mathcal{V}^{*}_{u}, \mathcal{Z}^{*}_{u}\}^{U}_{u=1}$.

Figure 4.3 is a scatter diagram showing the marginalized likelihood and K values (which are used widely for the measurement of the clustering) calculated from the results obtained when the proposed nested Gibbs sampler was applied to B1 and B1 with four types of noise. The values of K are explained in the Experiment section. The differences in the plots indicate the distinct speakers. This figure shows that the value of K is strongly correlated with the marginalized likelihood. Therefore, we can use the marginalized likelihood as a measure of the appropriateness of the models.



(a) $N^{Gibbs} = 1$ (b) $N^{Gibbs} = 5$

Figure 4.2: Logarithmic marginalized likelihood (LML) obtained using proposed nested Gibbs sampler, applied to A1 + station noise. Refer to Table 4.1 for the details of test set A1. Each figure shows results with a different sampling size N^{samp}. Eight lines correspond to results of eight trials using different random seeds.

4.4.4 Non-nested Gibbs sampler

Here, we briefly explain the conventional non-nested Gibbs sampler [31, 32] to give a comparison of this algorithm with the proposed nested Gibbs sampling. In this algorithm, fLVs and sLVs are separately sampled independent from each others, while they are simultaneously sampled in the nested Gibbs sampling. Therefore, the conventional Gibbs sampler has no nested structure as shown in Algorithm 4.

4.4.5 Simulated annealing

Conventional Gibbs sampler often suffers from a local optima problem because only a limited number of latent variables is updated in each step of sampling. To solve this problem, we introduce a simulated annealing (SA) [44]. This algorithm gradually reduces the SA temperature β in Eqs. 4.9 and 4.10 at each step of sampling. This enables the sampler to determine the latent variable in the early stages randomly. As a result, the system can effectively search the combination of latent variables. In all the following experiments, we introduce a geometrical scheduler defined as follows:

$$\beta^{t+1} \leftarrow \begin{cases} \gamma\beta^t, & if\ \beta^t > 1 \\ 1, & otherwise \end{cases} \tag{4.41}$$

Algorithm 4: Model estimation algorithm based on the conventional Gibbs sampling method.

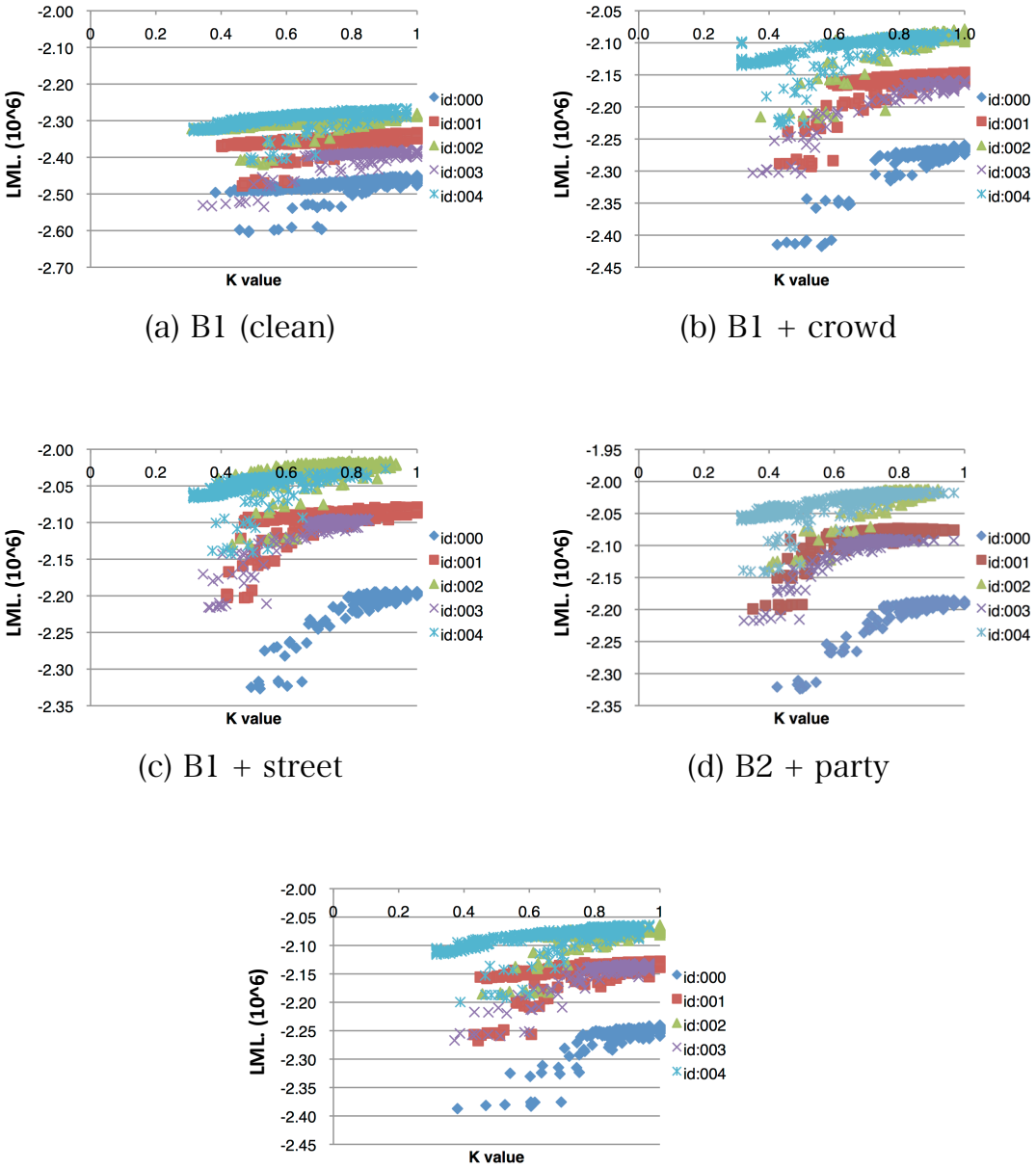
Initialization $\{z_u, v_{ut} : u = 1, \cdots, U, t = 1, \cdots, T_u\}$.;
repeat
  for *all segments* u *and frames* t **do**
    for *all components* j **do**
      Update $\gamma^{\beta}_{v_{ut}=j|z_u=i}$ by Eq. (4.39).
    end
    Draw the value of fLVs, v^*_{ut}, from its conditional posterior distribution, $v^*_{ut} \sim \gamma^{\beta}_{v_{ut}=\cdot|z_u=i}$
  end
  for *all utterances* u **do**
    for *all speakers* i **do**
      Update $\gamma^{\beta}_{z_u=i|\mathcal{V},\mathcal{Z}_{\setminus u}}$ by Eq. (4.36)
    end
    Draw segment-level latent variable (sLV), z^*_u, from its conditional posterior distribution, $z^*_u \sim \gamma^{\beta}_{z_u=i|\mathcal{V}^*_u}$
  end
until *some conditions are met*;

where γ is constant value which satisfies $0 < \gamma < 1$ We determined this value through pre-experiments.

4.5 Speaker clustering experiments

This section investigated the effectiveness of our model optimization methods at speaker clustering using the TIMIT [36] and CSJ [37] databases. The following four model estimation methods are evaluated:

- **n-Gibbs:** MCMC-based model estimation using the proposed nested Gibbs sampling method.
- **Gibbs:** MCMC-based model estimation using non nested Gibbs sampling where the fLVs and sLVs are sampled alternately [31, 32].
- **VB:** VB-based model estimation [45].
- **HAC-GMM:** hierarchical agglomerative clustering method. A GMM is estimated for each segment in a maximum likelihood manner. The similarity between segments is defined as the cross likelihood



(a) B1 (clean)

(b) B1 + crowd

(c) B1 + street

(d) B2 + party

(e) B2 + station

Figure 4.3: Logarithmic marginalized likelihood (LML) as a function of K value. Each plot shows the results obtained by applying the proposed n-Gibbs sampler to five different datasets (id: 000, 001, 002, 003, 004). Refer to Table 4.1 for the details of test set B1.

ratio between corresponding GMMs. The pair of segments with the greatest similarity is merged iteratively until the correct number of speakers is obtained [40].

4.5.1 Experimental setup

Datasets

All of the experiments were conducted using 11 evaluation sets obtained from TIMIT and CSJ. Table 4.1 lists the number of speakers and segments in the evaluation sets used. T1 and T2 were constructed using TIMIT. T1 corresponds to the core test set of TIMIT, which includes 192 segments from 24 speakers. T2 is the complete test set, which includes 1,152 segments from 144 speakers. In this case, there were no overlaps between T1 and T2. The remaining nine evaluation sets were constructed using CSJ as follows: all lecture speech in CSJ was divided into segment units based on the silence segments in their transcriptions, five speakers were then randomly selected, and five, 10, and 20 of their segments were chosen for A1, A2, and A3, respectively. In the same manner, 10 and 15 different speakers were randomly selected and five, 10, and 20 of their segments were used for B1 to B3 and C1 to C3, respectively. Five combinations of different speakers for each dataset were evaluated. The resulting clustering performance for each dataset was the average of these five combinations.

The speech data from TIMIT and CSJ are not corrupted by noise. In additional experiments, noisy speech data was used, which was created by overlapping each segment with four types of non-stationary noise (crowd, street, party, and station) selected from the noise database of the Japan Electronic Industry Development Association [46]. These noises were overlapped with each segment at a signal-to-noise ratio of about 10 dB. Speech data were sampled at 16 kHz and quantized into 16-bit data. We used 26-dimensional acoustic feature parameters, which comprised 12-dimensional mel-frequency cepstral coefficients (MFCCs) with log energy and their Δ parameters. The frame length and frame shift were 25 ms and 10 ms, respectively.

Table 4.1: Details of test set.

Test set	Number of Speakers	Number of Utterances	Average total Duration [min.]
T1	24	192	9.7
T2	144	1152	58.8
A1	5	25	2.8
A2	5	50	5.6
A3	5	100	11.1
B1	10	50	5.6
B2	10	100	11.3
B3	10	200	22.5
C1	15	75	13.0
C2	15	150	26.0
C3	15	300	51.8

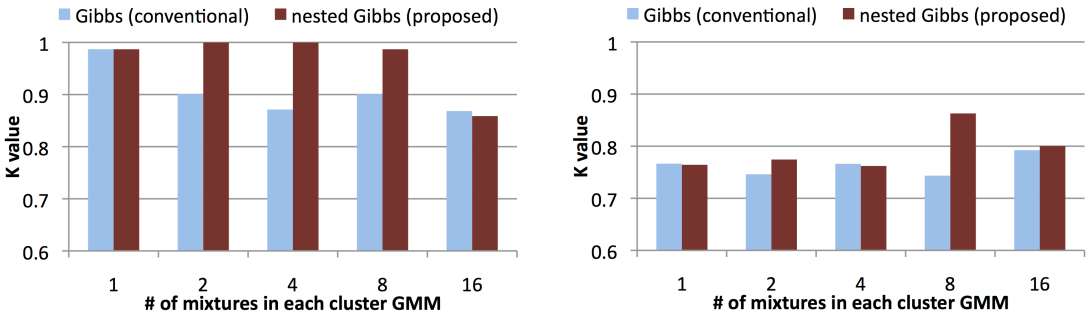
Table 4.2: K value for clean test sets.

Evaluation data	n-Gibbs	Gibbs	VB	HAC-GMM
T1 (spkr:24 utt:192)	**0.96**	0.84	0.74	0.88
T2 (spkr:144 utt:1152)	**0.74**	0.52	0.41	0.73
A1 (spkr:5 utt:25)	**1.00**	0.90	0.92	0.93
A2 (spkr:5 utt:50)	**0.99**	0.96	0.97	**0.99**
A3 (spkr:5 utt:100)	0.98	0.97	**0.99**	0.97
B1 (spkr:10 utt:50)	**0.98**	0.93	0.85	0.95
B2 (spkr:10 utt:100)	**0.98**	0.90	0.90	0.96
B3 (spkr:10 utt:200)	**0.98**	0.91	0.96	0.96
C1 (spkr:15 utt:75)	**0.97**	0.92	0.81	0.95
C2 (spkr:15 utt:150)	0.93	0.91	0.90	**0.96**
C3 (spkr:15 utt:300)	0.92	0.91	0.91	**0.95**

Evaluation conditions

We employed the average cluster purity (ACP), average speaker purity (ASP), and their geometric means (K value) as the speaker clustering evaluation criteria [8].

Sampling steps were iterated 100 times, which was sufficiently long for convergence in both the Gibbs and nested Gibbs sampling in all of the experimental conditions. We conducted the same experiment for eight times using different seeds. The marginalized likelihood described in Eq. 4.28 was calculated for each result and the result with the highest likelihood is selected from those obtained during the 100 iterations of



(a) A1 (b) A1 + crowd

Figure 4.4: *K values obtained by Gibbs and proposed nested Gibbs sampler applied on (a) clean (A1) and (b) noisy (A1 + crowd) speech.*

eight experiments.

The hyper-parameters in Eq. 4.25 were set as follows: $\mathbf{w}^{(0)} = \{\rho, \cdots, \rho\}$ for all components; $h^0 = \rho$ and $\mathbf{h}^{(0)} = \{\rho, \cdots, \rho\}$ for all clusters; $\eta^{(0)} = 1$ and $\xi^{(0)} = \rho$; $\boldsymbol{\mu}^{(0)} = \boldsymbol{\mu}(\mathcal{O})$ and $\boldsymbol{\Sigma}^{(0)} = \eta^0\boldsymbol{\Sigma}(\mathcal{O})$ where $\boldsymbol{\mu}(\mathcal{O})$ and $\boldsymbol{\Sigma}(\mathcal{O})$ were the mean vectors and covariance matrices estimated from the whole dataset, respectively. The value range for ρ was $\{1, 10, 100, 1000\}$. These parameters were determined using the development data set obtained from the CSJ dataset. We randomly initialized both the sLVs and fLVs .

4.5.2 Experimental results

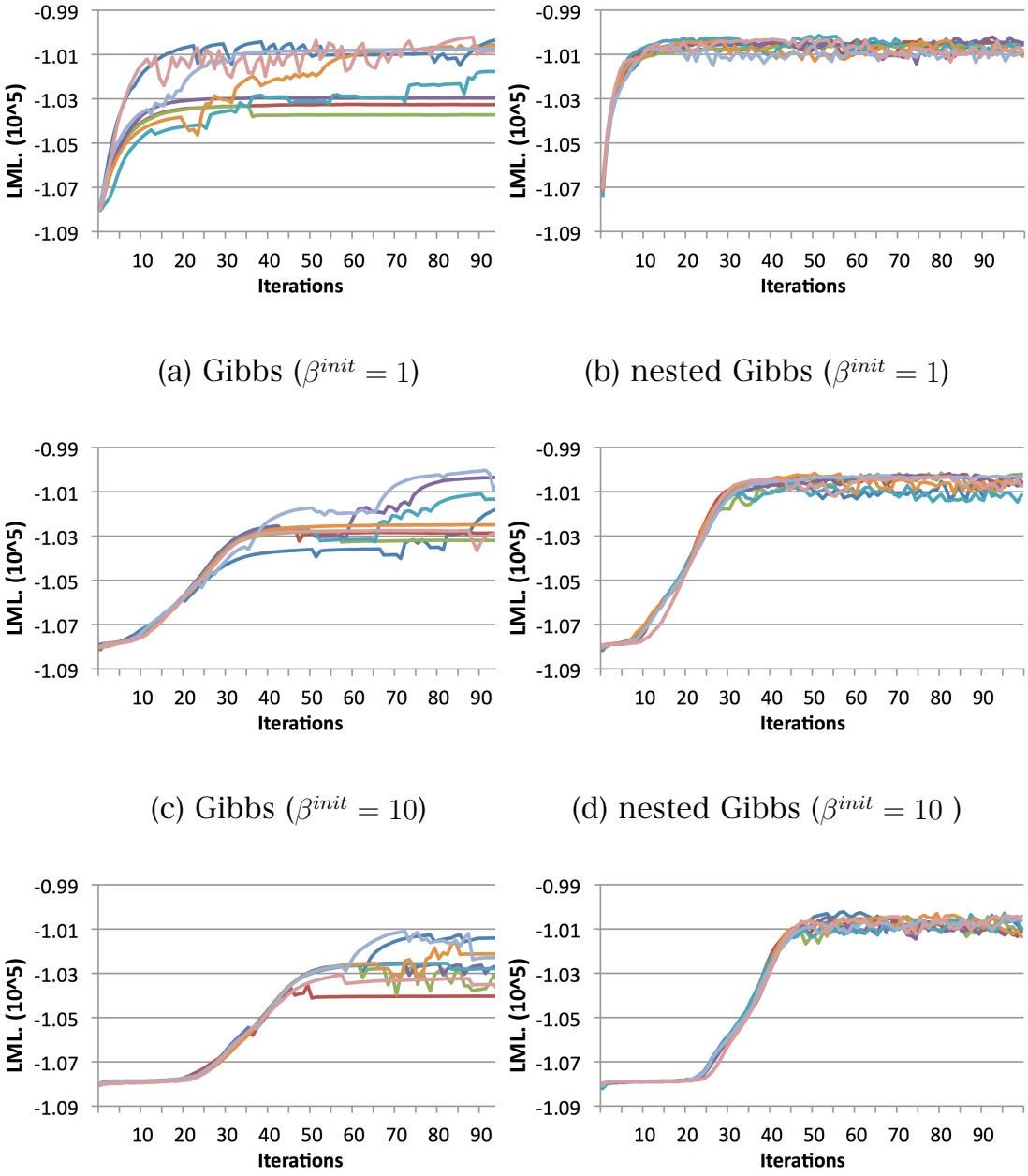
Comparison with the conventional Gibbs sampler

We evaluated conventional Gibbs sampling and the proposed nested Gibbs sampling method with different numbers of mixture components using both clean and noisy datasets. Figure 4.4 shows the K values obtained using the **Gibbs** and **n-Gibbs** samplers with different numbers of mixture components when they were applied to clean data (A1) and noisy data (A1 + crowd). We can see that the highest K value was obtained when one or two components were used for both the **Gibbs** and **n-Gibbs** samplers. This indicates that a small number of Gaussian distributions are sufficient to model each speaker's segments in either sampling method when clean data are used. However, in the case of noisy data, the nested Gibbs sampler performed best with eight components of mixtures, but the conventional Gibbs sampler with eight components

achieved worse results than the proposed method. This suggests that samples from noisy data follow a multi-modal distribution and that the proposed sampling method can represent this multi-modality. By contrast, the conventional Gibbs sampler could not model these complex data even with a large number of mixture components. Later, we will discuss the reason why the conventional Gibbs sampler degraded the K value for the noisy data set by using diagrams to show the convergence of the samplers.

Figure 4.5 shows the logarithmic marginalized likelihoods of the samples obtained using the conventional Gibbs and proposed nested Gibbs sampling methods when applied to A1 with different SA temperatures [44]. The eight lines in these figures represent the results obtained from eight trials with different seeds. We can see that no trial converged to a unique distribution without SA (i.e., $\beta^{init} = 1$) when a conventional Gibbs sampler was used. Introducing a higher temperature ($\beta^{init} = 30$) offered some protection from divergence, but large variations still remained, as shown in Figures 4.5 (c) and (e). These results indicate that the conventional Gibbs sampler was often trapped by a local optimum. However, in the case of the nested Gibbs sampler, the likelihoods converged after only 20 iterations at most, and all of the trials converged to almost the same result, even when we did not use the SA method (i.e., $\beta^{init} = 1$). These results indicate the greater effectiveness of the proposed sampling method.

Tables 4.2 and 4.3 list the K values obtained using each method for clean and noisy speech data, respectively. These tables demonstrate that the nested Gibbs sampler outperformed the conventional Gibbs sampler irrespective of the evaluation sets, under clean and noisy conditions. These results imply that the proposed method can model data drawn from both single and multi-modal distributions, which the conventional Gibbs sampler was unable to calculate.



(a) Gibbs ($\beta^{init} = 1$)

(b) nested Gibbs ($\beta^{init} = 1$)

(c) Gibbs ($\beta^{init} = 10$)

(d) nested Gibbs ($\beta^{init} = 10$)

(e) Gibbs ($\beta^{init} = 30$)

(f) nested Gibbs ($\beta^{init} = 30$)

Figure 4.5: *Logarithmic marginalized likelihood (LML) obtained by Gibbs and nested Gibbs with simulated annealing applied on A1. Each figure shows result with different initial temperature β^{init}. Eight lines correspond to the results of eight trials with different seeds.*

Comparison with the VB-based method and hierarchical agglomerative method

The K values determined using the VB-based and agglomerative methods are also listed in Tables 2 and 3. The results obtained by the proposed method were equal or superior to those with the conventional VB-based (**VB**) methods using both the clean and noisy datasets. In particular, the proposed method obtained substantially better performance when the data were very scarce (e.g. A1, B1, C1, T1, and T2). This implies that nested Gibbs sampling-based estimation can adequately estimate the cluster structure from limited data, which is generally difficult to achieve. In fact, the VB-based method cannot model such limited data. To evaluate the effectiveness of a fully Bayesian approach, we also compared the proposed method with the conventional hierarchical agglomerative method (**HAC-GMM**). The proposed method also outperformed the HAC-GMM in most conditions.

Computational cost

We now consider the computational cost based on two features: the number of iterations until convergence and the computation required for each epoch.

The T-1 dataset (i.e., 24 speakers and 192 segments; 9.7 minutes in total) was used for this experiment. The **VB** approach required about 14.8 seconds on average for one epoch and 12 iterations until it converged (i.e., Real-time factor (RTF) of about 0.0031) when an Intel Xeon 3.00 GHz processor was used. However, the proposed nested Gibbs sampling method required about 41.4 seconds on average for one epoch and about 63 iterations until the maximum logarithmic marginalized likelihood was obtained (i.e., RTF of about 0.0450), whereas the conventional Gibbs sampling method only required about 1.58 seconds and about 17 iterations until the maximum logarithmic marginalized likelihood was obtained (i.e., RTF of about 0.0005). Figure 4.5 (a) shows the logarithmic marginalized likelihood obtained when the nested Gibbs

sampler was applied to dataset A1. We can see that the chain of samples obtained using the nested Gibbs sampler converged within 100 iterations at most. Compared with the conventional Gibbs sampler, the nested Gibbs sampler required more iterations and computations while it obtains substantially better performance.

In fact, the computational cost of the nested Gibbs sampler will increase drastically as the number of segments increases because many iterations are needed during the sampling process. However, the sampling of fLVs can be parallelized, because the posterior distribution of fLVs is calculated independently of the segments. Thus, we can reduce the computational time by using multi-threading technology. Fortunately, sampling procedure for frame-level latent variables easy to parallelize because the calculation of the posterior for fLVs w.r.t an segment is independent from those of the other segments. We, therefore, can parallelize these procedure using the parallelized techniques such as general purpose graphical processing unit (GPGPU) or multi-threading technologies.

4.6 Summary

In this chapter, we formalized the segment-generative model as a mixture of Gaussian mixture models (GMMs) by modeling each cluster by a GMM. Derived model has a specific structure called mixture of mixtures and has a more powerful representation ability for capturing multimodality. We also formulated a novel model estimation method called nested Gibbs sampler o estimate the hierarchical structure of MoGMMs. The proposed nested Gibbs sampler can efficiently avoid local optimum solutions due to its nested sampling procedure, where the structure of its elemental mixture distributions are sampled jointly. We showed that the proposed method can estimate models accurately for speech segments drawn from complex multi-modal distributions, wheres the results obtained by the conventional Gibbs sampler-based method were trapped in local optima. The proposed method also outperformed the conventional agglomerative approach in most conditions.

Table 4.3: K value for noisy test sets. Four types of noise (crowd, street, party, and station) are overlapped with speech of nine datasets.

Evaluation data	n-Gibbs	Gibbs	VB	HAC-GMM
A1 + crowd (spkr:5 utt:25)	**1.00**	0.90	0.82	0.95
A2 + crowd (spkr:5 utt:50)	**0.99**	0.96	0.95	0.97
A3 + crowd (spkr:5 utt:100)	**0.99**	0.97	0.99	0.95
B1 + crowd (spkr:10 utt:50)	**0.97**	0.92	0.83	0.93
B2 + crowd (spkr:10 utt:100)	**0.97**	0.94	0.91	0.92
B3 + crowd (spkr:10 utt:200)	**0.93**	0.88	0.92	0.89
C1 + crowd (spkr:15 utt:75)	**0.99**	0.96	0.79	0.96
C2 + crowd (spkr:15 utt:150)	**0.99**	0.95	0.91	0.94
C3 + crowd (spkr:15 utt:300)	**0.96**	0.90	0.90	0.92
A1 + street (spkr:5 utt:25)	**0.86**	0.74	0.69	0.79
A2 + street (spkr:5 utt:50)	**0.78**	0.66	0.69	0.77
A3 + street (spkr:5 utt:100)	**0.86**	0.72	0.84	0.75
B1 + street (spkr:10 utt:50)	**0.84**	0.75	0.62	0.79
B2 + street (spkr:10 utt:100)	**0.75**	0.68	0.66	0.73
B3 + street (spkr:10 utt:200)	**0.72**	0.62	0.71	0.71
C1 + street (spkr:15 utt:75)	**0.77**	0.67	0.60	0.75
C2 + street (spkr:15 utt:150)	**0.68**	0.60	0.61	**0.68**
C3 + street (spkr:15 utt:300)	0.68	0.62	**0.71**	0.68
A1 + party (spkr:5 utt:25)	**0.97**	0.87	0.88	0.95
A2 + party (spkr:5 utt:50)	0.99	0.93	**1.00**	0.87
A3 + party (spkr:5 utt:100)	**1.00**	0.92	0.99	0.96
B1 + party (spkr:10 utt:50)	**0.98**	0.88	0.83	0.95
B2 + party (spkr:10 utt:100)	**0.96**	0.86	0.88	0.95
B3 + party (spkr:10 utt:200)	**0.96**	0.89	0.90	0.92
C1 + party (spkr:15 utt:75)	**0.98**	0.93	0.81	0.94
C2 + party (spkr:15 utt:150)	**0.94**	0.91	0.87	0.92
C3 + party (spkr:15 utt:300)	**0.92**	0.90	0.90	0.90
A1 + station (spkr:5 utt:25)	**0.92**	0.86	0.77	0.87
A2 + station (spkr:5 utt:50)	0.86	0.76	**0.90**	0.85
A3 + station (spkr:5 utt:100)	0.84	0.75	0.86	**0.87**
B1 + station (spkr:10 utt:50)	**0.89**	0.79	0.69	0.86
B2 + station (spkr:10 utt:100)	0.84	0.77	0.76	**0.86**
B3 + station (spkr:10 utt:200)	**0.81**	0.75	0.81	**0.81**
C1 + station (spkr:15 utt:75)	**0.89**	0.79	0.69	0.84
C2 + station (spkr:15 utt:150)	**0.89**	0.74	0.77	0.80
C3 + station (spkr:15 utt:300)	0.81	0.73	**0.83**	**0.83**

Chapter 5

Speaker clustering based on spectral information

5.1 Introduction

In the previous chapter, segments are clustered by estimating the MoGMMs for the whole dataset. This approach, however, requires a large number of computations that exponentially increases with the number of frame-wise observations. In this chapter, we explore another approach based on i-vectors to segment clustering.

5.2 i-vector-based approach

Recently, a simplified GMM-based approach has been proposed in the speaker recognition community and achieves state-of-the-art performance in NIST speaker recognition evaluation tasks. In this approach, each segment is modeled via a maximum a posteriori (MAP) adaptation of the prior GMM. Then, the mean "supervector" is generated by concatenating all of the mean vectors of the Gaussian components. Fig 5.1 depict the supervector obtained by adapting prior GMM to each segment.

Here, the supervectors are experimentally shown to be distributed in far lower-subspace than their original space. In order to obtain these low-dimensional supervectors, the following factor analysis model is applied to supervectors:

$$\boldsymbol{m}_u = \boldsymbol{m}_0 + \boldsymbol{T}\boldsymbol{x}_u, \tag{5.1}$$

where $\boldsymbol{m}_0$ denotes the supervector of u-th segment, $\boldsymbol{m}_0$ denotes the prior GMM supervector, T is a rectangular low-rank matrix representing the

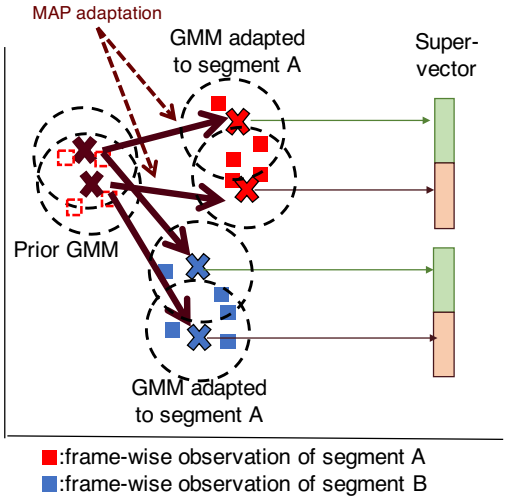


Figure 5.1: Depiction of super-vector-based approach.

total variability space, and $\boldsymbol{x}_u$ is a random vector with a standard Gaussian prior distribution $\mathcal{N}(0, \mathbf{I})$. Then, the i-vector is derived as a MAP mean of $\boldsymbol{x}_u$ given an observed utterance $\mathbf{O}_u$.

Here, we present a derivation of the posterior expectation of the i-vector using the sufficient statistics of prior GMM. The zeroth and centralized first-order statistics of the segment $\mathbf{o}_t, t = 1 \cdots, T_u$ are derived as follows:

$$\begin{cases} N_j(u) = \sum_{t=1}^{T_u} \gamma_t(j), \\ \tilde{\mathbf{F}}_j(u) = \sum_{t=1}^{T_u} \gamma_t(j)(\mathbf{o}_t - \boldsymbol{\mu}_j), \end{cases} \tag{5.2}$$

where $j = 1, ..., L$ is the index of the Gaussian component, $\gamma_t(j) = P(c|\mathbf{o}_u t, \boldsymbol{\Theta})$ denotes the posterior probability of the frame-wise observation $\mathbf{o}_{ut}$ to the j-th mixture component of the prior GMM, and $\boldsymbol{\mu}_j$ denotes the mean vector of the j-th mixture component of the prior GMM. Let $\mathbf{N}(u)$ be the $LD \times LD$ diagonal matrix whose j-th diagonal blocks are $N_j(u) \cdot \mathbf{I}$, and let $\tilde{F}(u)$ be the $CF \times 1$ supervector obtained by concatenating $\tilde{\mathbf{F}}_j(u)$. Then, the posterior expectation of the i-vector is derived as follows:

$$\mathrm{E}[\boldsymbol{x}_u] = \boldsymbol{G}_u \boldsymbol{T}^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathrm{ivec}}^{-1} \boldsymbol{F}_u, \tag{5.3}$$

where

$$\boldsymbol{G}_u = (\boldsymbol{I} + \boldsymbol{T}^{\mathrm{T}} \boldsymbol{\Sigma}_{\mathrm{ivec}}^{-1} \boldsymbol{N}_u \boldsymbol{T})^{-1}. \tag{5.4}$$

Both $\boldsymbol{T}$ and $\boldsymbol{\Sigma}_{\mathrm{ivec}}$ in Eq. 5.4 are estimated via the EM algorithm. A full explanation of this EM algorithm can be found in [47].

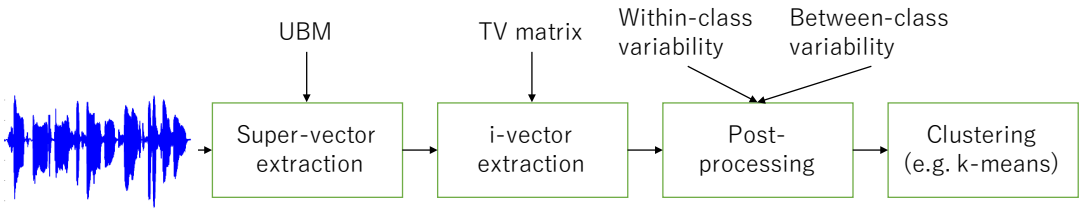


Figure 5.2: Diagram of the i-vector system for segment clustering.

i-vectors obtained by Eq. 5.4 contain speaker information as well as intra-speaker variability derived from difference in channels, phoneme contexts, and environmental noises. Such nuisance information is proven to be successfully eliminated by using LDA and WCCN [47]. Then, the cosine distance is applied to measure the similarity between a pair of i-vectors, which is extracted as follows:

$$\cos(\boldsymbol{w}_i, \boldsymbol{w}_j) = \frac{\boldsymbol{w}_i^{\mathrm{T}} \boldsymbol{w}_j}{||\boldsymbol{w}_i|| \cdot ||\boldsymbol{w}_j||}, \tag{5.5}$$

where $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$ denote the i-vectors extracted from the i-th and j-th utterances, respectively. Fig. 5.2 denotes a diagram of i-vector system for segment clustering.

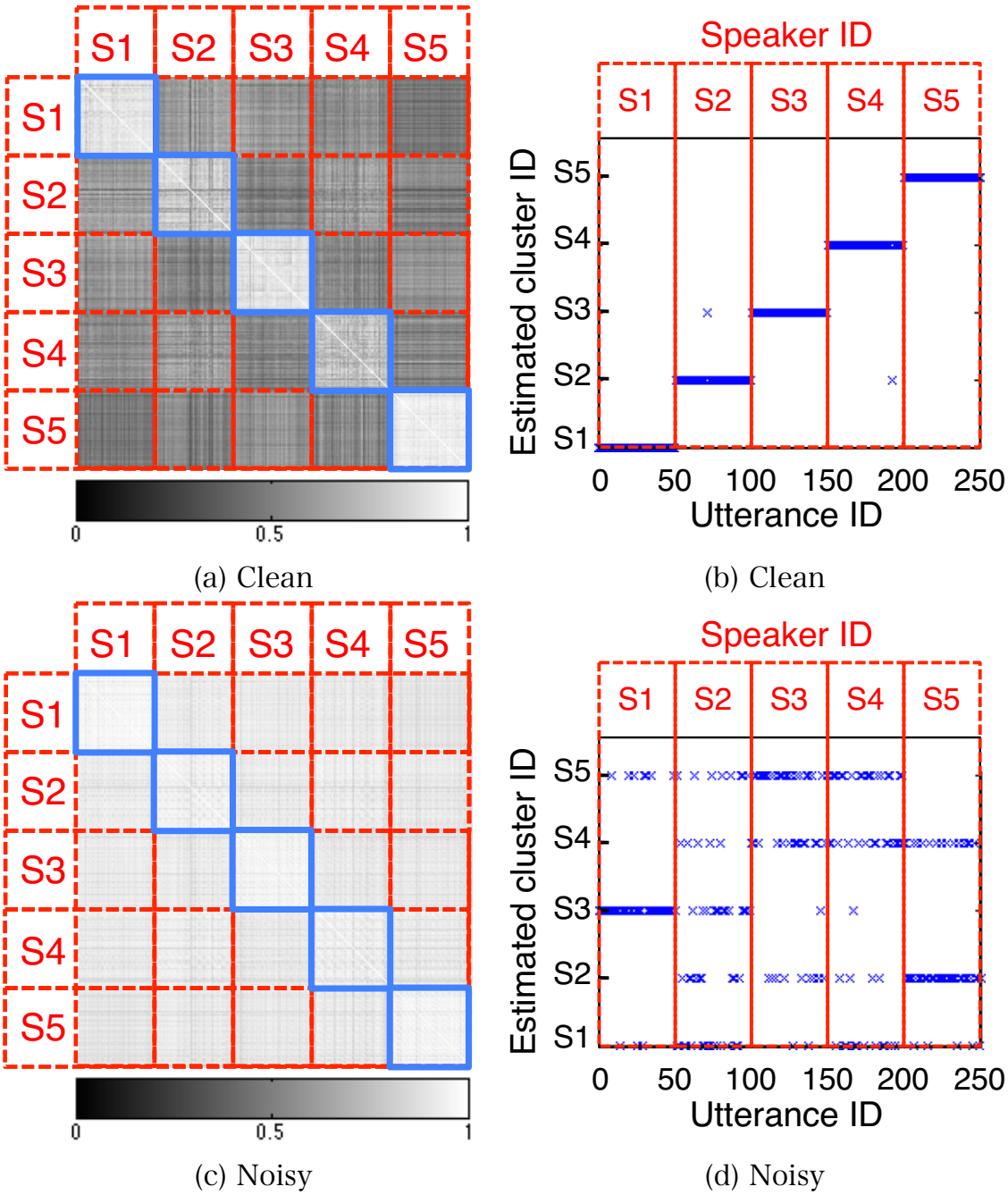
The i-vector approach has also been applied to speaker clustering problems [48, 49]. In speaker clustering, an utterance is represented by an i-vector, and k-means clustering based on the cosine similarity is applied for clustering these vectors. Spectral clustering was evaluated as an alternative to k-means clustering [50, 51, 49]. Generally, spectral clustering works better than k-means clustering because of its capability of classifying data with a type of manifold embedding. However, the authors of [49] concluded that the simple k-means clustering algorithm provides a sufficiently high accuracy because i-vectors are linearly separable on the unit hypersphere, hence, the cosine distance is a valid measurement. The assumption that i-vectors are separably distributed on the unit hypersphere is not always true under noisy conditions because the i-vector contains not only speaker information but also noise information, which cannot be perfectly eliminated by front-end processing methods such as LDA and WCCN. In this chapter, we evaluate the

effectiveness of spectral clustering for various types of noise. The experimental comparisons using lecture data demonstrated that spectral clustering provides a significant improvement in speaker clustering, especially for nonstationary noise. The results obtained in the present study can be useful for developing noise-robust speaker clustering and diarization systems.

The rest of this chapter is organized as follows. In Section 5.3, we present some examples of the failure of the conventional i-vector-based approach, and the purposes for employing spectral clustering are discussed. Section 5.4 provides a brief explanation of spectral clustering and indicates the effectiveness of this approach for speech data corrupted by noise. In Section 5.5, experimental comparisons are conducted to demonstrate the effectiveness of spectral clustering for noisy speech. Section 5.6 summarizes this chapter.

5.3 i-vectors under mismatched condition

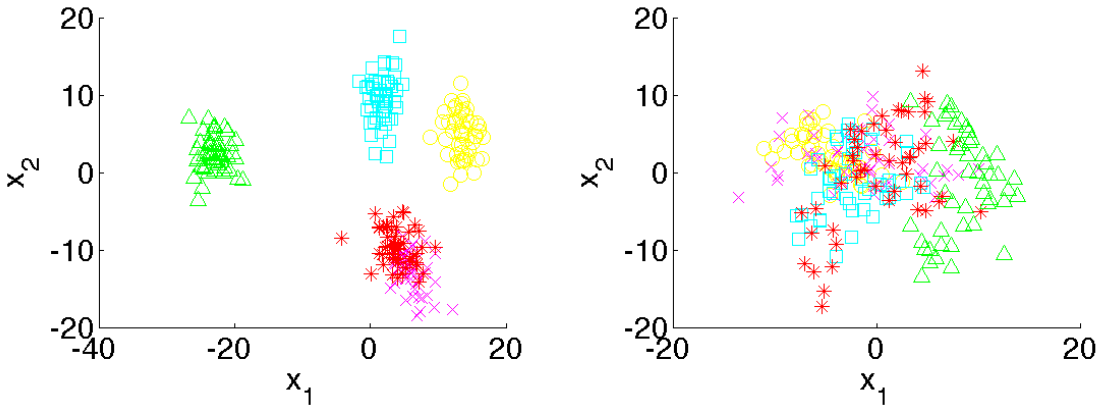
We compare the similarity matrices calculated from clean and noisy speech utterances to investigate the effect of noise on the cosine similarity score extracted from i-vectors. In this experiments, WCCN, LDA, and TV matrices are trained clean segments. Fig. 5.3 (a) and (c) depict the similarity matrix of 500 utterances from five speakers in a clean and noisy environments respectively. In both figures, five rectangles drawn with dashed lines, S1 to S5, correspond to each speaker's utterances, and regions with blue dashed lines indicate the similarity between utterances from the same speaker. From these figures, we can see that i-vectors from the same speaker always have a large similarity compared with ones from the different speakers. Figs. 5.3 (b) and (d) show the speaker clustering results obtained by k-means clustering for clean and noisy utterances, respectively. These results show that i-vector-based similarity fails to capture the speaker similarity for noisy segments. We applied k-means clustering on those clean and noise disturbed segments. In these figures, vertical axis shows the estimated cluster ID of each utterance. These results also indicate that i-vector-based similarity fails



(a) Clean (b) Clean

(c) Noisy (d) Noisy

Figure 5.3: Similarity matrix obtained from (a) clean and (c) noisy utterances. Clustering result obtained by applying k-means clustering on i-vectors from (b) clean and (d) noisy utterances.



(a) Clean (b) Noisy

Figure 5.4: The i-vectors of five speakers after LDA/WCCN projection onto two-dimensional space. Each color corresponds to speaker.

to capture the similarities of noise disturbed utterances. Therefore, i-vector-based similarity is insufficient to measure the speaker similarity for noisy utterances.

Further analysis is performed by investigating the distribution of i-vectors. Figs. 5.4 (a) and (b), respectively, depict distributions of i-vectors obtained from clean and noisy utterances from five speakers. Each i-vector is projected onto two-dimensional space by LDA. In each figure, colors correspond to speaker ID. From these figures, we can see that the samples from noise corrupted segment spread over a large and complex region, whereas ones from clean segment spread over a relatively small region.

5.4 Spectral clustering

Spectral clustering on i-vector-based similarity is carried out to handle noise corruption in speaker clustering. Here, we describe a brief explanation of spectral clustering and how it works for speech corrupted by noise.

Spectral clustering is a top-down approach to determine the optimal assignment of utterances to clusters; it assumes any pair of samples in the same cluster has high similarity, while those from a different cluster should have low similarity. Assume that an indicator vector

Algorithm 5: Algorithm of Ng-Jordan-Weiss spectral clustering [52].

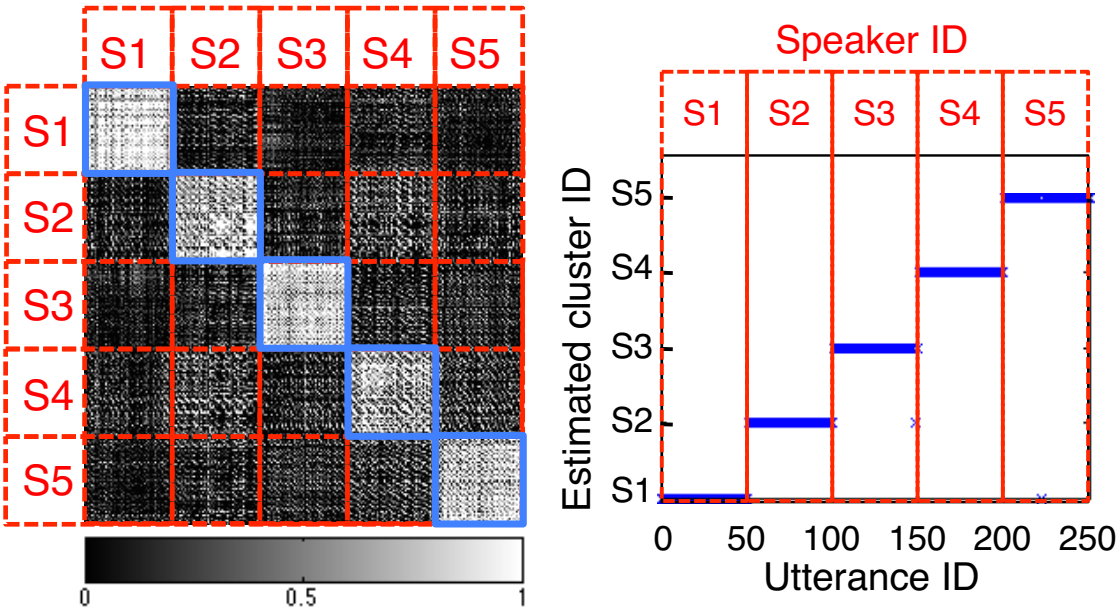
1: Calculate cosine-based distance between all pair of i-vectors $D(\boldsymbol{x}_i, \boldsymbol{x}_j) = 1 - \cos(\boldsymbol{x}_i, \boldsymbol{x}_j), \forall i, j$.
2: Calculate adjacency matrix $\boldsymbol{W} \in \mathbb{R}_+^{n \times n}$, where $(\boldsymbol{W})_{ij} = \exp\{-D(\boldsymbol{x}_i, \boldsymbol{x}_j)\}$ for $i \neq j$ and zero otherwise.
3: Calculate the diagonal matrix $\boldsymbol{D}$ whose (i, i)-th components is sum of i-th row of $\boldsymbol{W}$ (i.e. $(\boldsymbol{D})_{ii} = \sum_{j=1}^{n} w_{ij}$), and construct the graph Laplacian $\mathrm{L} = \boldsymbol{I} - \boldsymbol{D}^{-\frac{1}{2}} \boldsymbol{W} \boldsymbol{D}^{-\frac{1}{2}}$.
4: Select $\boldsymbol{t}_1, \cdots, \boldsymbol{t}_K$, K smallest eigenvectors of L and form $\boldsymbol{T} = [\boldsymbol{t}_1, \cdots, \boldsymbol{t}_K] \in \mathbb{R}^{n \times K}$.
5: Normalize each row of $\boldsymbol{T}$ to have unit length (i.e. $\{\hat{\boldsymbol{T}}\}_{ij} = \{\boldsymbol{T}\}_{ij} / (\sum_k t_{ik}^2)^{1/2}$)
6: Cluster row vectors of $\hat{\boldsymbol{T}}$ via cosine similarity-based k-means clustering.

$\boldsymbol{t}_i = [t_{i1}, \cdots, t_{ij}, \cdots, t_{in}]^{\mathrm{T}} \in \mathbb{R}^n$ represents the assignment of the j-th sample to the i-th cluster. A component t_{ij} is equal to $1/\sqrt{(\boldsymbol{D})_{ii}}$ if the j-th sample is assigned to the i-th cluster and equal to zero otherwise. For any indicator vector $\boldsymbol{t}_i$, we have

$$\boldsymbol{t}_i^{\mathrm{T}} \boldsymbol{L} \boldsymbol{t}_i = \frac{1}{2} \sum_{j=1}^{n} \sum_{j'=1}^{n} w_{jj'} (t_{ij} - t_{ij'})^2, \tag{5.6}$$

where $w_{jj'}$ denotes the similarity between the j-th and j'-th segments and n denotes the number of segments. Eq. 5.6 indicates that $\boldsymbol{t}_i^{\mathrm{T}} \boldsymbol{L} \boldsymbol{t}_i$ will be small if two samples with large similarity (i.e., $w_{jj'}$ is large) have similar coordinates (i.e., t_{ij} and $t_{ij'}$ are close). This implied that the indicator vector $\boldsymbol{t}_i$ obtained by minimizing Eq. 5.6 under the constraint of all indicator vectors being orthogonal can indicate a valid clustering result in which pairs of utterances from the same speaker have a large similarity and those from the different speakers have small similarity. The solution of this minimization problem is obtained as the K smallest eigenvectors of $\boldsymbol{L}$.

Fig. 5.5 (a) depicts the similarity matrix of noise-corrupted segments calculated using the smallest 20 normalized eigenvectors of the Laplacian matrix $\boldsymbol{L}$. This figure shows that these eigenvectors are reasonable features to measure the similarity in speakers because the similarity thus obtained between the same speaker's utterances is higher than that from



(a) (b)

Figure 5.5: (a) Similarity matrix calculated from normalized eigenvectors of the Laplacian matrix of noisy utterances of five speakers. (b) Clustering result obtained by k-means clustering using the eigenvectors-based features.

different speakers. Fig. 5.5 (b) depicts the clustering result obtained by k-means clustering on the eigenvectors-based features. This figure shows that spectral clustering can perfectly cluster the utterances even when they are corrupted by noise.

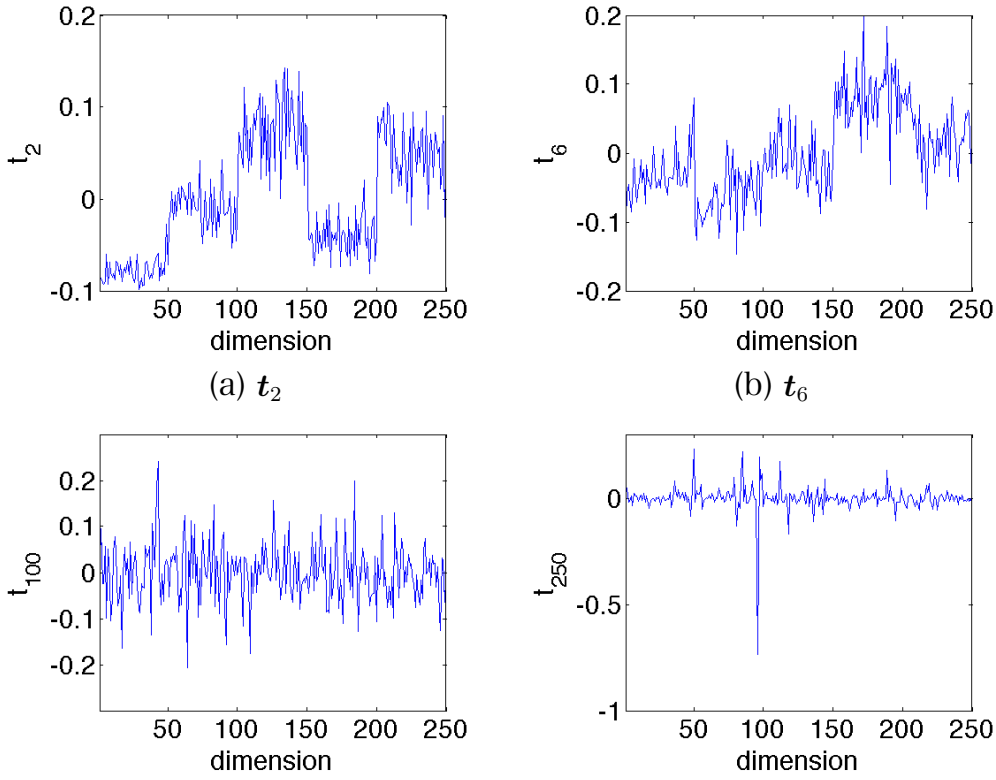
Next, we present a more detailed analysis of the effectiveness of eigenvector-based features for measuring the speaker similarity under the noise conditions. The spectral clustering utilizes the K smallest eigenvectors of the Laplacian matrix as features of each utterance. Therefore, we have

$$\boldsymbol{L} = \boldsymbol{T}\boldsymbol{\Lambda}\boldsymbol{T}^{\mathrm{T}} = \sum_{i=1}^{n} \lambda_i \boldsymbol{t}_i \boldsymbol{t}_i^{\mathrm{T}}, \tag{5.7}$$

where $\boldsymbol{\Lambda} = [\lambda_1, \cdots, \lambda_N]$ and $\boldsymbol{T} = [\boldsymbol{t}_1 \cdots, \boldsymbol{t}_N]$ denote the eigenvalues and corresponding eigenvectors of the Laplacian matrix, respectively. The Laplacian matrix of the noisy utterances can be factorized into two matrices: the similarity matrix primarily obtained from speech signals and that from noise signals. Here, there exists a clear pattern that intra-speaker similarity is relatively higher than inter-speaker similarity. The similarly in terms of noise signals, on the other hand, is consistently

low because there are no correlations between noise signals. Considering this fact and that $0 = \lambda_1 < \lambda_2, \cdots < \lambda_n$, smaller eigenvectors should correspond to a pattern representing speaker similarity, whereas larger eigenvectors should correspond to a pattern representing similarity in noise.

Fig. 5.6 depicts the second, sixth, 100th, and 250th smallest eigenvectors of the Laplacian matrix of noisy utterances. This figure shows that the smaller eigenvectors restore speakers' similarity patterns while the larger ones restore the similarity patterns of noise.



(a) $\boldsymbol{t}_2$ (b) $\boldsymbol{t}_6$

(c) $\boldsymbol{t}_{100}$ (d) $\boldsymbol{t}_{250}$

Figure 5.6: The (a) second, (b) sixth, (c) 100th and (d) 250th smallest eigenvectors of the Laplacian matrix calculated from noisy utterances.

5.5 Speaker clustering experiments

Experimental comparisons were performed to demonstrate robustness of spectral clustering against acoustic mismatches of training data. For

that purpose, the following three methods were evaluated under various types of noise.

- **GMM-HAC**: Agglomerative clustering in which each cluster is modeled as a Gaussian mixture model (GMM). Similarity between clusters is defined as a cross likelihood ratio between these GMMs [40].
- **IV-KMEANS**: k-means clustering using cosine distance between i-vectors [48, 49].
- **IV-SC**: Spectral clustering using cosine distance between i-vectors.

The present experiments were conducted using the corpus of spontaneous Japanese (CSJ) [37].

5.5.1 Experimental setups

Noisy conditions

The clean and noisy speech utterances from CSJ were used for evaluation. Note that speech data from CSJ are basically uncorrupted by noise. The clean evaluation sets were constructed as follows. All of the lecture speech in CSJ were divided by the utterance on the basis of silence. Then, 10 speakers were randomly selected. Finally, their 50 utterances were randomly selected. Each utterance is from both the same and different lectures. Four combinations of different speakers yielded four evaluation sets and the resulting performance was the average over those four sets.

In addition, noisy speech data were developed by overlapping each utterance with seven types of noise at the signal-to-noise ratio (SNR) of about 0 dB. The noise includes four types of environmental noise (Crowd, Party, Street, and Station) sampled from JEIDA noise database [53] and three types of background music sampled from RWC music databases [54]. Note that crowd and party noises are stationary while street and station noises are non-stationary. The other experimental setups for evaluating noisy speech were the same as for clean speech. The evaluation criteria was the K value, which is the geometric mean of the average speaker purity and average cluster purity [8].

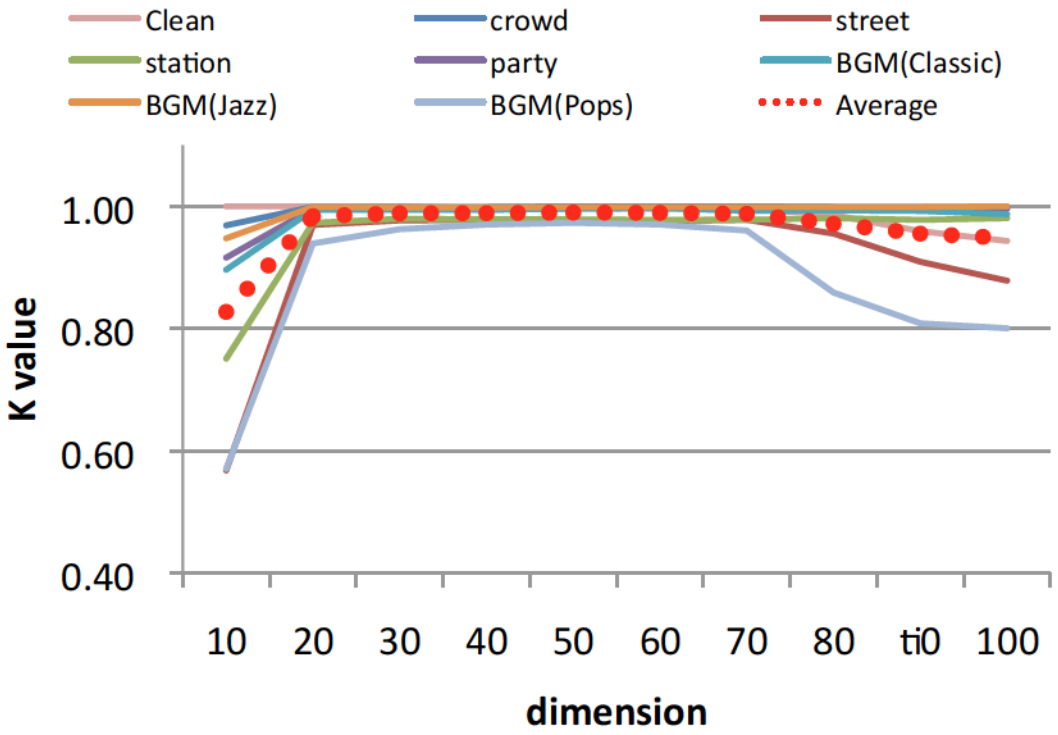


Figure 5.7: Clustering accuracy as a function of number of eigenvectors.

5.5.2 Front-end processing

Acoustic feature parameters consisted of 12-dimensional mel-frequency cepstral coefficients (MFCCs) plus log-energy and their delta parameters, yielding a 26 dimensional vector for every 10 ms. A gender-independent GMM of 128 Gaussians with diagonal covariance matrices was trained on the speech data taken from the Japanese newspaper article sentence (JNAS) [55] and Continuous Speech Corpus for Research (ASJ-JIPDEC) databases [56]. Those speech data were overlapped with four types of noise (air conditioner, car, factory, and plant) from JEIDA noise database and the total variability, LDA, and WCCN matrices were trained on those noisy data. The 150-dimensional i-vectors were extracted and finally projected onto 100-dimensional vectors.

5.5.3 Experimental results

Number of Eigenvectors

In ideal conditions, samples from different clusters are infinitely far apart, yielding always zero-similarity between those samples. In this case, the eigenvectors of the Laplacian matrix are consistent to the optimal indicator vectors and the optimal number of eigenvectors coincides the number of clusters. However, this assumption is not always true in the noise conditions because the similarity between distinct speakers' utterances can be large. Figure 5.7 depicts the clustering accuracy as a function of the number of eigenvectors. Eight solid lines describe the clustering accuracy obtained from spectral clustering in clean and seven noise conditions. This figure shows that the highest clustering accuracy was achieved when the number of eigenvectors was larger than that of speakers. This was noticeable particularly in the noise conditions because further eigenvectors are required to recover the Laplacian matrix from the noisy utterances. The experiments below used 50 eigenvectors providing the highest performance for all conditions.

Clustering Accuracy

Table 5.1 lists the K values obtained using three clustering methods for clean and noisy speech data. This result demonstrates that the clean utterances were almost perfectly clustered, irrespective of methods. However, the agglomerative clustering (GMM-HAC) and k-means clustering on i-vectors (IV-KMEANS) consistently deteriorated clustering accuracy in noisy conditions and yielded significant degradation particularly in the non-stationary and BGM (pops) noise conditions. Note that IV-KMEANS outperformed GMM-HAC in the stationary noise conditions but not in the non-stationary noise conditions. The i-vector-based similarity is therefore sufficient to handle the stationary noise corruptions but insufficient for the non-stationary noise. In contrast, spectral clustering on i-vectors (IV-SC) worked in both stationary and non-stationary noise conditions with small or almost no degradation in clustering performance compared with the clean conditions. We also evaluated these

Table 5.1: K values obtained from Speaker clustering experiment. Average duration of each utterance is about 20 seconds.

Environment		GMM-HAC	IV-KMEANS	IV-SC
Clean		0.955	**1.000**	**1.000**
Stationary noise	Crowd	0.906	0.997	**1.000**
	Party	0.907	0.958	**0.999**
Non-Stationary noise	Street	0.425	0.540	**0.976**
	Station	0.591	0.591	**0.979**
BGM	Classic	0.769	0.930	**0.996**
	Jazz	0.821	0.989	**0.999**
	Pops	0.301	0.383	**0.973**

Table 5.2: K values obtained from Speaker clustering experiment. Average duration of each utterance is about 10 seconds.

		GMM-HAC	IV-KMEANS	IV-SC
Clean		0.900	**1.000**	**1.000**
Stationary noise	Crowd	0.672	0.809	**0.981**
	Party	0.727	0.752	**0.964**
Non-Stationary noise	Street	0.225	0.331	**0.876**
	Station	0.398	0.470	**0.820**
BGM	Classic	0.355	0.604	**0.964**
	Jazz	0.467	0.789	**0.983**
	Pops	0.193	0.263	**0.665**

methods on relatively short utterances. Here, the average duration is about 10 seconds. The result is shown in Table 5.2, and demonstrates that IV-SC also outperformed GMM-HAC and IV-KMEANS for these short utterances.

5.6 Summary

This chapter examines the efficiency of spectral clustering on i-vector-based representation for speech corrupted by noise. For noisy segment, i-vectors seriously are corrupted because the condition of training data seriously differs from the testing condition. This mismatch makes i-vectors unreliable and clustering accuracy is seriously deteriorated. We showed that spectral clustering can yield robustness against noise by

non-linear projection. This is contributed by the fact that spectral clustering is effective for removing noisy factors from noise corrupted similarity matrix. From speaker clustering experiments under noisy and mismatched conditions, we can see that spectral clustering yielded significant improvement from conventional methods, such as agglomerative clustering and k-means clustering, under non-stationary noise conditions.

Chapter 6

Conclusion and future directions

The goal of this thesis was to establish segment-wise clustering frameworks that can robustly work in a situation where segments are corrupted by nuisance information such as background noise. In order to achieve this purpose, we tried to leverage the prior knowledge of a model structure using Bayesian estimation.

In Chapter 2, we formalized a segment-generative model and provided an overview of the conventional hierarchical agglomerative approaches. We then showed that HAC approaches have several intrinsic drawbacks and introduced an alternative approach based on mixture modeling.

In Chapter 3, we formalized the segment-generative model as a mixture of Gaussian distributions by modeling each cluster as a single Gaussian distribution. We extended this model to a non-parametric Bayesian speaker modeling based on SO-DPMM to make it possible to estimate the number of speakers in model-based speaker clustering. The experimental comparison demonstrated that the proposed method was successfully applied to speaker clustering on practically large-scale data and outperformed the existing HAC method.

In Chapter 4, we proposed a novel method for estimating a mixture-of-mixtures model. The proposed nested Gibbs sampler can efficiently avoid local optimum solutions owing to its nested sampling procedure, where the structure of its elemental mixture distributions are sampled

jointly. We showed that the proposed method can estimate models accurately for speech utterances drawn from complex multimodal distributions, wheres the results obtained by the conventional Gibbs-sampler-based method were trapped in local optima. The proposed method also outperformed the conventional agglomerative approach in most conditions. In this work the proposed MoGMMs can build a hierarchical model from multi-level data that comprise frame-wise observations. Some types of real-world data also has the same kind of structure, such as images comprising a set of pixels. In future research, we plan to apply MoGMMs to the other tasks such as image clustering problem. Nonparametric Bayesian approaches have recently been attracting attention as methods for selecting optimal model structures. For example, the nested Dirichlet process mixture model [57] provides a model selection solution for our MoGMMs. As shown in Chapter 3, the mixture model can be extended to a nonparametric version to estimate the optimal number of speakers. Actually nonparametric Bayesian version of a mixture-of-mixture model was proposed in [58, 59], and was demonstrated that it was effective in estimating the number of speakers However, this model was based on the conventional Chinese restaurant process, and we employed the conventional Gibbs sampling method, which is readily trapped in local optima. In future research, we plan to develop a nested Gibbs-sampling-based method for such nonparametric Bayesian models.

Chapter 5 proposed i-vector-based spectral clustering that effectively removes the noise-derived component from the similarity matrix between i-vectors. The proposed approach was evaluated for the speaker clustering problem for various types of noise and yielded significant gains from conventional agglomerative and i-vector-based k-means clustering. This work assumes that the correct number of clusters is known. Several attempts have been made to discover the number of clusters equal to the optimal number of eigenvectors on the basis of the largest gradient of eigenvalues for clean speech data [51, 49]. However, the present work showed that to achieve a high accuracy of speaker clustering under noisy conditions, the number of eigenvectors should be

greater than the number of actual clusters. This implies that the eigenvalue-based method is no more applicable to the estimation of the number of clusters for noisy speech data, and an alternative approach is required. An attempt will be made as future work to estimate the optimal number of clusters using a more sophisticated manner, e.g., self-tuning spectral clustering [60] and denoised kernel spectral clustering [61]. Besides, selecting the optimal number of eigenvectors was also important to obtain a robust feature in the embedded space. In this research, we simply selected a sufficiently large number of eigenvectors greater than the number of clusters. However, we also showed that the clustering accuracy gradually decreased when the number of selected eigenvectors was too large. This result indicated that the selecting the optimal number of eigenvectors is also important as well as the number of clusters. Recently, some attempts have been applied to select the optimal eigenvectors in [62, 63, 64]. Most of these approaches, however, are limited to simulation data. We, therefore, attempt to apply these techniques to our real world data. Improving the robustness of segment representation is an another direction to improve the clustering performance. Multi-condition training [65] and probabilistic latent discriminative analysis scoring [66] will be utilized to construct more robust similarity matrix.

Lastly, we discuss to the future direction of the segment-wise clustering problem. Through this thesis, we assumed that all segments are previously obtained by using an existing segmentation method such as the voice activity detection method. Considering that we apply our approach to realistic applications, however, these boundaries are not always given. Actually, in the NIST speaker diarization evaluation, the segmentation performance is evaluated along with the clustering performance. Therefore, our approach needs to estimate them simultaneously. One direction to achieve this purpose is to introduce an iterative approach that iteratively solves the segmentation and clustering tasks [67]. We are planing to apply our MoGMMs- and i-vector-based approaches to this framework.

Another line of future work is to combine our approach with the deep learning technique [68]. Recently, deep learning technologies have

achieved state-of-the-art performances in various fields. One of the key factors of their success is that they can optimize the whole network, including feature extraction. One simple approach to combine our approach with the deep technology is to apply our approach in the feature space spanned by the deep network. Besides, we are planning to integrate our approach by defining a new optimization function that incorporates clustering and feature extraction. Based on the progress of deep learning technologies, a new segment-wise task called segmental embedding has attracted a considerable amount of attention in many fields such as exemplar-based speech recognition [69], text-query search [70, 71] and language acquisition [72]. In those problems, segments are embedded in a fixed dimensional space and clustered within it. It is of interest to combine our segmental approaches with these embedding frameworks.

Acknowledgments

Firstly, I would like to express my sincere gratitude to my supervisor, Professor Tetsunori Kobayashi, for his continuous support of my Ph.D. study. His guidance helped me through all my research and my writing of this thesis. I also would like to thank Professors Yasuo Matsuyama, Daichi Mochihashi, and Tetsuji Ogawa, who acted as vice supervisors. I would particularly like to thank Professor Ogawa for his help. His teachings covered not only research activities but also general aspects such as social postures, technical writing, presentation, and the proper attitudes of a researcher. Without his consistent guidance and invaluable help, this thesis would never have been possible.

I would like to thank all the concerned members of the NTT Communication Science (CS) Laboratories at NTT Corporation. The main research topic of this thesis originated in the corroborative research with the CS laboratory. The months I spent there were an irreplaceable experience. All the discussions we had there were very stimulating and made me decide to be a researcher. I would like to offer special thanks to Doctor Shinji Watanabe (currently at the Mitsubishi Electric Research Laboratories). His insightful comments derived from his deep knowledge of speech and Bayesian theory always lighted my way. Above all, he is always teaching me how to conduct research and how to live as a researcher. He always shows me his kindness and consideration such as in giving me many opportunities to interact with researchers who are active on the front lines. His teaching has covered so many aspects that I can not list all.

I would like to express my gratitude to the faculty in our laboratory. Professor Shinya Fujie at Chiba Institute of Technology taught me a lot of things during my bachelor period. He taught me a lot about the fun of research. Professor Tsuneo Nitta gave me insightful comments and encouragement from his deep insight into linear algebra. Doctor

Kazuya Ueki gave me insightful comments and suggestions through his insight into image processing. I thank my fellow labmates especially Motoi Omachi and Yohei Shiraishi for the stimulating discussions for the sleepless nights.

Last but not the least, I would like to thank all of my friends and my family for their continuous support throughout my life. Words cannot express how grateful I am to my mother and my grandmother, who always prepared dinner for me no matter how late I came home. Finally, I would like to give special thanks to my father, who supported me morally and financially for a long time.

January 2017

Acknowledgments in Japanese

まず本研究の着手及び方針につきまして，多くのご指導，ご助言をいただいた早稲田大学小林哲則教授に心より感謝いたします．この論文をまとめるにあたり，また，これまでの研究における様々な場合において小林教授の的確なご助言に助けていただきました． 副査としてご指導いただいた早稲田大学松山泰男教授，小川哲司准教授，統計数理研究所持橋大地准教授に深く感謝致します． 特に，小川准教授には厚くご指導賜りましたのでこの場を借りて深く御礼申し上げます． 小川准教授には研究に直接関連することのみならず，プレゼンテーションや論文執筆のノウハウに関するご指導， そして研究者としての心構えなど，今後の研究者人生の指針となるたくさんのご助言をいただきました． この数え切れないほどのご支援が無ければこの研究は完成しませんでした．

日本電信電話株式会社 NTT コミュニケーション科学基礎研究所の関係者の皆様へ感謝いたします． 本論文の主要な成果は本研究所で行った共同研究が基礎となっています． 実際に研究所に滞在した期間は短い期間でしたがその経験は私にとってかけがいのないものです． ここで経験した刺激的な議論が，私に研究者として生きていくことを決意させたきっかけの一つだったと思っております． 中でも特に熱心に御指導頂いた渡部晋治博士 (現 Mitsubishi Electric Research Laboratories) に深く感謝致します． 渡部博士からは音声やベイズ理論をはじめとした様々な分野に渡る高い見識から多大なご助言を頂きましたが， 何より，研究者としての心構え，そして研究者としてどうあるべきかを教えていただきました． また様々な場面で多くのチャンスを頂いたり，第一線でご活躍している研究者の方々をご紹介頂いたり等， 渡部博士から頂いたご支援は多岐に渡り，この場で全てを列挙することはできないほどです．

小林研究室の教員の皆様にも心から感謝しております． 千葉工業大学藤江真也准教授には学部生時代に特にお世話になりました． 当時，まだ右も左もわからなかった自分に研究の楽しさを教えていただきました． 植木一也助教には画像等のマルチメディアデータに関する深い見識から的確なご指摘を頂きました． 新田恒雄客員教授には，部分空間法や線形代数に関する深い見

識から研究のアイデアを頂きました．また，大町基氏や白石洋平氏をはじめとして研究室の同僚や後輩にも大変刺激を受けました．

最後に，心の支えとなっていただいた友人たち，そして私の人生を通じ経済的，精神的に援助し続けて頂いた家族に深く感謝いたします．どんなに帰宅が遅くなっても食事を用意して待っていてくれた母と祖母，そして長くに渡り支えて頂いた父に深く感謝致します．

2017年 1月

Appendices

A.1 Formulations of distributions

This section defines the distributions described in chapters 3 and 4.

Data:

U is the total number of segments.
T_u: is the total number of frames in the u-th segment.
n_i is the number of segments in the i-th cluster.
n_{ij}: is the number of frames in the j-th component in the i-th cluster.
$\mathbf{o}_{ut}$: is the t-th frame-wise observation in u-th segment.
z_u: is the u-th segment level latent variable.
v_{ut}: is the t-th frame level latent variable in u-th segment.

Parameters:

h_i: is the weight / concentration parameter of the i-th cluster.
w_{ij}: is the weight parameter of the i-th component in the i-th cluster.
$\boldsymbol{\mu}_{ij}$: is the mean vector of the j-th component in the i-th cluster.
$(\eta_{ij})^{-1}\boldsymbol{\Sigma}_{ij}$: is the covariance matrix of the j-th component in the i-th cluster.

Hyper-parameters:

D: is the dimensionality of data.
K: is the number of component in MoGMMs of data.
S: is the number of clusters.
$\tilde{h}_i, h^0$: are the posterior and prior weight / concentration parameter of the i-th cluster.
$\tilde{w}_{ij}, w_j^0$: are the posterior and prior weight / concentration parameter of the j-th component in the i-th cluster.
$\tilde{\xi}_{ij}, \xi_j^0$: are the posterior and prior relative precision of $\tilde{\boldsymbol{\mu}}_{ij}$ compared with the data.
$\tilde{\eta}_{ij}, \eta_j^0$: are the posterior and prior degrees of freedom of precision.
$\tilde{\boldsymbol{\mu}}_{ij}, \boldsymbol{\mu}_j^0$: are the posterior and means of $\boldsymbol{\mu}_{ij}$.
$(\tilde{\eta}_{ij})^{-1}\tilde{\boldsymbol{\Sigma}}_{ij}, (\tilde{\eta}_j^0)^{-1}\tilde{\boldsymbol{\Sigma}}_j^0$: are the posterior and prior means of $\boldsymbol{\Sigma}_{ij}$ and $\boldsymbol{\Sigma}_j^0$.

A.1.1 Likelihood functions

The diagonal Gaussian distribution for the t-th frame-wise observation in the u-th segment is written as follows:

$$\begin{aligned} p(\mathbf{o}_{ut}|\mathbf{\Theta}_{ij}) &= \mathcal{N}(\mathbf{o}_{ut}|\boldsymbol{\mu}_{ij}, \mathbf{\Sigma}_{ij}) \\ &= \prod_{d=1}^{D} \frac{1}{(2\pi)^{1/2}(\sigma_{ij,d})^{1/2}} \exp\left\{-\frac{(o_{ut,d}-\mu_{ij,d})^2}{2\sigma_{ij,d}}.\right\} \end{aligned} \tag{A.1}$$

The multinomial distributions for the u-th segment-level latent variable and t-th frame-level latent variable in the u-the segment are written as follows:

$$p(z_u|\mathbf{h}) = \mathcal{M}(z_u|\mathbf{h}) = \prod_{i=1}^{S} h_i^{\delta(z_u,i)}, \tag{A.2}$$

$$p(v_{ut}|\mathbf{w}_i) = \mathcal{M}(v_{ut}|\mathbf{w}_i) = \prod_{j=1}^{K} w_{ij}^{\delta(v_{ut},j)}, \tag{A.3}$$

where $\delta(a, b)$ denotes the Kronecker delta, which is 1 if $a = b$ and 0 otherwise.

A.1.2 Prior distributions

The Dirichlet distribution is written as

$$P(\mathbf{h}) = \mathcal{D}(\mathbf{h}|\mathbf{h}^0) = \frac{\Gamma(h^0)}{S \cdot \Gamma(h^0)} \prod_i h_i^{\frac{h^0}{S}-1}, \tag{A.4}$$

$$P(\mathbf{w}_i) = \mathcal{D}(\mathbf{w}_i|\mathbf{w}_i^0) = \frac{\Gamma(w_i^0)}{S \cdot \Gamma(w_i^0)} \prod_j w_{ij}^{w_i^0-1}. \tag{A.5}$$

The expectations of $\log h_i$ and $\log w_{ij}$ are respectively described as follows [18]:

$$\langle \log h_i \rangle = \psi(h_i^0) - \psi\left(\sum_i h_i^0\right) \tag{A.6}$$

$$\langle \log w_{ij} \rangle = \psi(w_{ij}^0) - \psi\left(\sum_j w_{ij}^0\right) \tag{A.7}$$

The Gaussian-Gamma distribution for the parameter of the j-th component in the i-th Gaussian distribution is written as

$$\begin{aligned}
p(\boldsymbol{\Theta}_{ij}) &= \mathcal{N}G(\boldsymbol{\mu}_{ij}, \boldsymbol{\Sigma}_{ij}|\xi^0, \eta^0, \boldsymbol{\mu}^0, \boldsymbol{\Sigma}^0) \\
&= \prod_{i,j} \mathcal{N}(\boldsymbol{\mu}_{ij}|\boldsymbol{\mu}^0, (\xi^0)^{-1}\boldsymbol{\Sigma}_{ij}) \prod_d \mathcal{G}(\sigma_{dd}|\eta^0, \sigma^0_{dd}) \\
&= \prod_{ij}\prod_d \frac{\xi^0}{(2\pi)^{1/2}(\sigma_{ij,dd})^{1/2}} \exp\left\{-\frac{\xi^0(\mu_{ij,d}-\mu^0_d)^2}{2\sigma_{ij,dd}}\right\} \\
&\quad \cdot \frac{1}{\Gamma(\eta^0)}(\sigma^0_{dd})^{\frac{\eta^0}{2}}\sigma_{ij,dd}^{-\eta^0+1}\exp\left(-\frac{\sigma^0_{dd}}{2\sigma_{ij,dd}}\right) \\
&= \prod_{ij} \frac{(\xi^0)^{\frac{D}{2}}\left(\prod_d \sigma^0_{dd}\right)^{\frac{\eta^0}{2}}}{(2\pi)^{D/2}\Gamma(\eta^0)^{\frac{D}{2}}} \left(\prod_d \sigma_{ij,dd}\right)^{-\eta^0+\frac{1}{2}} \\
&\quad \cdot \exp\left\{-\sum_d \frac{1}{2\sigma_{ij,dd}}\left(\xi^0(\mu_{ij,d}-\mu^0_d)^2+\sigma^0_{dd}\right)\right\} \\
&= \prod_{i,j} \frac{1}{Z(\xi^0, \eta^0, \boldsymbol{\mu}^0, \boldsymbol{\Sigma}^0)} \left(\prod_d \sigma_{ij,dd}\right)^{-\eta^0+\frac{1}{2}} \\
&\quad \cdot \exp\left\{-\sum_d \frac{1}{2\sigma_{dd}}\left(\xi^0(\mu_{ij,d}-\mu^0_d)^2+\sigma^0_{dd}\right)\right\}
\end{aligned} \tag{A.8}$$

where,

$$Z(\xi^0, \eta^0, \boldsymbol{\mu}^0, \boldsymbol{\Sigma}^0) = \frac{(2\pi)^{\frac{D}{2}}\Gamma(\eta^0)^D}{(\xi^0)^{\frac{D}{2}}\left(\prod_d \sigma^0_{dd}\right)^{\frac{\eta^0}{2}}}. \tag{A.9}$$

A.2 VB posterior calculation Posterior distribution

In this section, we derive the variational posterior distributions of MoGMMs in Chapter 4.

A.2.1 Latent variables

First, we explain how to derive the posterior distributions of $\mathcal{V}$ and $\mathcal{Z}$ described as Eqs. 4.17 and 4.19.

As we described in section 4.3.1, the posterior distributions of $\mathcal{V}$ and $\mathcal{Z}$ are derived as follows:

$$q(\mathcal{V}|\mathcal{Z}) \propto \exp\Big(\Big\langle \log p(\mathcal{O},\mathcal{V},\mathcal{Z},\boldsymbol{\Theta})\Big\rangle_{q(\boldsymbol{\Theta})}\Big), \tag{A.10}$$

$$q(\mathcal{Z}) \propto \exp\Big(\Big\langle \log p(\mathcal{O},\mathcal{V},\mathcal{Z},\boldsymbol{\Theta})\Big\rangle_{q\mathcal{V})q(\boldsymbol{\Theta})}\Big). \tag{A.11}$$

Removing any terms that are independent of $\mathcal{V}$ from Eq. A.10, we have

$$\begin{aligned}\log q(\mathcal{V}|\mathcal{Z}) &\propto \exp\Big(\Big\langle \log p(\mathcal{O},\mathcal{V},\mathcal{Z},\boldsymbol{\Theta})\Big\rangle_{q(\boldsymbol{\Theta})}\Big)\\ &= \exp\Big(\Big\langle \log p(\mathcal{O},\mathcal{V},\mathcal{Z},\boldsymbol{\Theta})\Big\rangle_{q(\mathbf{w})q(\boldsymbol{\mu},\boldsymbol{\Sigma})}\Big)\\ &= \Big\langle \log p(\mathcal{V}|\mathcal{Z},\mathbf{w})\Big\rangle_{q(\mathbf{w})} + \Big\langle \log p(\mathcal{O}|\mathcal{Z},\mathcal{V},\boldsymbol{\mu},\boldsymbol{\Sigma})\Big\rangle_{q(\boldsymbol{\mu},\boldsymbol{\Sigma})}.\end{aligned} \tag{A.12}$$

Substituting for the two conditional distributions on the right-hand side with Eqs. 4.3 and 4.4, and removing any terms that are independent of $\mathcal{V}$, we have

$$\begin{aligned}\log q(\mathcal{V}|\mathcal{Z}) &\propto \sum_{u,i}\delta(z_u,i)\sum_{t,j}\delta(v_{ut},j)\Big(\langle\log w_{ij}\rangle_{q(w_{ij})} + \frac{1}{2}\sum\nolimits_d\langle\log\sigma_{ij,d}\rangle_{q(\sigma_{ij,d})}\\ &\quad -\frac{D}{2}\log 2\pi - \frac{1}{2}\sum\nolimits_d\langle\frac{(o_{ut,d}-\mu_{ij,d})^2}{\sigma_{ij,d}}\rangle_{q(\mu_{ij,d}|\sigma_{ij,d})}\Big)\\ &\propto \sum_{u,i}\delta(z_u,i)\sum_{t,j}\delta(v_{ut},j)\log\gamma^*_{v_{ut}=j|z_u=i;\hat{\boldsymbol{\Theta}}}.\end{aligned} \tag{A.13}$$

where, we have defined

$$\begin{aligned}&\gamma^*_{v_{ut}=j|z_u=i;\tilde{\boldsymbol{\Theta}}}\\ &\quad\triangleq \exp\Big(\langle\log w_{ij}\rangle_{q(w_{ij})} + \frac{1}{2}\sum\nolimits_d\langle\log\sigma_{ij,d}\rangle_{q(\sigma_{ij,d})}\\ &\qquad -\frac{D}{2}\log 2\pi - \frac{1}{2}\sum\nolimits_d\langle\frac{(o_{ut,d}-\mu_{ij,d})^2}{\sigma_{ij,d}}\rangle_{q(\mu_{ij,d}|\sigma_{ij,d})}\Big).\end{aligned} \tag{A.14}$$

In the same manner, removing any terms that are independent of $\mathcal{Z}$ from Eq. 4.15, we have

$$\log q(\mathcal{Z}) \propto \Big\langle \log p(\mathcal{Z}|\mathbf{h})\Big\rangle_{q(\mathbf{h})} + \Big\langle \log p(\mathcal{O}|\mathcal{Z},\mathcal{V},\boldsymbol{\mu},\boldsymbol{\Sigma})\Big\rangle_{p(\mathcal{V})p(\boldsymbol{\mu},\boldsymbol{\Sigma})}. \tag{A.15}$$

Substituting for the two conditional distributions on the right-hand side, and again removing any terms that are independent of $\mathcal{Z}$, we have,

$$\log q(\mathcal{Z}) = \sum_u\sum_i \delta(z_u,i)\log\gamma_{z_u=i;\tilde{\boldsymbol{\Theta}}}, \tag{A.16}$$

where, we have defined

$$\gamma^*_{z_u=i;\tilde{\boldsymbol{\Theta}}} \triangleq \exp\left(\langle \log h_i \rangle_{q(h_i)} + \sum_t \log \sum_j \gamma^*_{v_{ut}=j|z_u=i;\tilde{\boldsymbol{\Theta}}}\right). \quad \text{(A.17)}$$

We can determine the posterior distribution of an fLV by normalizing Eq. A.17 as follows:

$$q(z_u = i) = \frac{\gamma^*_{z_u=i;\tilde{\boldsymbol{\Theta}}}}{\sum_i \gamma^*_{z_u=i;\tilde{\boldsymbol{\Theta}}}} \triangleq \gamma_{z_u=i}. \quad \text{(A.18)}$$

To keep the notation uncluttered, we have omitted the $\tilde{\boldsymbol{\Theta}}$ on $\gamma_{z_u=i}$.

For the multinomial distribution, we have the standard result that

$$\left\langle \delta(z_u, i) \right\rangle_{q(z_u=i)} = \gamma_{z_u=i}. \quad \text{(A.19)}$$

A.2.2 VB posterior of model parameters

Next, we derive the variational posterior distribution of model parameters $\boldsymbol{\Theta}$. As we described in 4.3.1, the variational posterior distributions of $\boldsymbol{\Theta}$ is derived as

$$q(\boldsymbol{\Theta}) \propto \exp\left(\left\langle \log p(\mathcal{O}, \mathcal{V}, \mathcal{Z}, \boldsymbol{\Theta}) \right\rangle_{q(\mathcal{V},\mathcal{Z})}\right). \quad \text{(A.20)}$$

In the same manner as the derivation of latent variables, removing any terms that are independent of $\boldsymbol{\Theta}$ from Eq. A.20, we have

$$\begin{aligned}
\log q(\boldsymbol{\Theta}) &\propto \exp\left(\left\langle \log p(\mathcal{O}, \mathcal{V}, \mathcal{Z}, \boldsymbol{\Theta}) \right\rangle_{q(\mathcal{V},\mathcal{Z})}\right) \\
&\propto \log p(\mathbf{h}) + \sum_i p(\mathbf{w}_i) + \sum_{i,j}\sum_d \log p(\mu^0_d, \sigma^0_d) \\
&\quad + \left\langle \log p(\mathcal{Z}|\mathbf{h}) \right\rangle_{q(\mathcal{Z})} + \sum_i \left\langle \log p(\mathcal{V}|\mathcal{Z}, \mathbf{w}_i) \right\rangle_{q(\mathcal{V},\mathcal{Z})} \\
&\quad + \sum_{u,t}\sum_{i,j} \left\langle z_u = i \right\rangle_{q(\mathcal{Z})} \left\langle v_{ut} = j \right\rangle_{q(\mathcal{V}|\mathcal{Z})} \sum_d \log \mathcal{N}(o_{ut,d}|\mu_{ij,d}, \sigma_{ij,d}).
\end{aligned} \quad \text{(A.21)}$$

Again, removing any terms that are independent of $\mathbf{h} = \{h_i\}_{i=1}^S$ from Eq. A.21, we obtain the variational posterior of weight of clusters $\mathbf{h}$ as follows:

$$\begin{aligned}
\log q(\mathbf{h}) &\propto \log p(\mathbf{h}) + \left\langle \log p(\mathcal{Z}|\mathbf{h}) \right\rangle_{q(\mathcal{Z})} \\
&\propto \sum_i \log h_i^{h^0-1} + \sum_{u,i} \left\langle \log h_i^{\delta(z_u,i)} \right\rangle_{q(\mathcal{Z})} \\
&= \sum_i \log h_i^{h^0-1} + \sum_i \log h_i^{\sum_u \gamma_{z_u=i}},
\end{aligned} \quad \text{(A.22)}$$

Here, we have used Eq. A.19. Taking the exponential of both sides, we obtain $q(\mathbf{h})$ as a Dirichlet distribution

$$\begin{aligned} q(\mathbf{h}) &= \prod_i h_i^{h^0+\sum_u \gamma_{z_u=i}-1} \\ &\propto \mathcal{D}(\tilde{\mathbf{h}}), \end{aligned} \tag{A.23}$$

where we defined

$$\begin{aligned} \tilde{h}_i &= h^0 + \sum_u \gamma_{z_u=i} \\ &= h^0 + \sum_u c_i. \end{aligned} \tag{A.24}$$

Using Eq. A.23 and the definition of Dirichlet distribution described in Eq. A.7, Eq. 4.21 in section 4.3.1 is derived as follows:

$$\langle \log h_i \rangle_{q(h_i)} = \psi(\tilde{h}_i) - \psi(\textstyle\sum_i \tilde{h}_i), \tag{A.25}$$

In the same way, removing corresponding terms that are independent of $\mathbf{w}_i$ from Eq. A.21, we can derive the variational posterior of weight of the j-th component in the i-th cluster as follows:

$$\begin{aligned} \log q(\mathbf{w}_i) &\propto \sum_j \log p(w_{ij}) + \sum_{i,j}\sum_{u,t} \gamma_{z_u=i}\gamma_{v_{ut}=j|z_u=i} \log w_{ii} \\ &\propto \sum_j \log w_j^{w_i^0+\sum_{u,t} \gamma_{z_u=i}\gamma_{v_{ut}=j|z_u=i}-1} \qquad \text{(A.26)} \\ q(\mathbf{w}_i) &= \prod_j w_{ij}^{w_i^0+\sum_{u,t} \gamma_{z_u=i}\gamma_{v_{ut}=j|z_u=i}-1} \\ &\propto \mathcal{D}(\tilde{\mathbf{w}}_i), \qquad \text{(A.27)} \end{aligned}$$

where we defined

$$\begin{aligned} \tilde{w}_{ij} &= w^0 + \sum_{u,t} \gamma_{z_u=i}\gamma_{v_{ut}=j|z_u=i} \\ &= w^0 + n_{ij}. \end{aligned} \tag{A.28}$$

Using Eq. A.27 and definition of Dirichlet distribution, Eq.4.22 in section 4.3.1 is derived as follows:

$$\langle \log w_{ij} \rangle_{q(w_{ij})} = \psi(\tilde{w}_{ij}) - \psi(\textstyle\sum_j \tilde{w}_{ij}). \tag{A.29}$$

In order to derive the variational posterior of $\mu_{ij,d}$ and $\sigma_{ij,d}$, we start with removing corresponding terms that are independent of $\mu_{ij,d}\sigma_{ij,d}$ to give

$$\begin{aligned}
&\log q(\mu_{ij,d}, \sigma_{ij,d}) \\
&\propto \log \mathcal{N}(o_{ut}|\mu_{ij}, \sigma_{ij,d}) + \mathcal{N}G(\xi^0, \eta^0, \mu^0, \sigma^0) \\
&\quad + \sum_{u,t}\sum_{i,j} \Big\langle z_u = i \Big\rangle_{q(\mathcal{Z})} \Big\langle v_{ut} = j \Big\rangle_{q(\mathcal{V}|\mathcal{Z})} \log \mathcal{N}(o_{ut}|\mu_{ij}, \sigma_{ij,d}) \\
&\quad = \sum_{i,j}\sum_{d} -\frac{\xi^0(\mu_{ij,d} - \mu^0_{j,d})^2}{\sigma_{ij,d}} - \frac{1}{2}\log \sigma_{ij,d} + \frac{1}{2}\sigma^0_{j,d}{\sigma_{ij,d}}^{-1} \\
&\quad + \left(\sum_{u,t} \gamma_{z_u=i}\gamma_{v_{ut}=j|z_u=i}\right)\sigma_{ij,d} + \eta^0\sigma^0_{j,d} + \sum_{u,t}\gamma_{z_u=i}\gamma_{v_{ut}=j|z_u=i}\frac{(o_{ut} - \mu_{ij,d})^2}{\sigma_{ij,d}}.
\end{aligned} \tag{A.30}$$

Using the definition of Gauss-Gamma distribution, we can express $\log q(\mu_{ij,d}, \sigma_{ij,d})$ as $\log q(\mu_{ij,d}|\sigma_{ij,d}) + \log q(\sigma_{ij,d})$. Here, the variational posterior of $\mu_{ij,d}$ conditioned on $\sigma_{ij,d}$ is derived as follows:

$$\begin{aligned}
&\log q(\mu_{ij,d}|\sigma_{ij,d}) \\
&= -\frac{1}{2}{\mu_{ij,d}}^2[\xi^0 + \sum_{u,t}\gamma_{z_u=i}\gamma_{v_{ut}=j|z_u=i}] + [\xi^0\mu^0_{j,d} + \sum_{u,t}\gamma_{z_u=i}\gamma_{v_{ut}=j|z_u=i}o_{ut}]\sigma_{ij,d} \\
&= -\frac{{\mu_{ij,d}}^2[\xi^0 + n_{ij}] - \mu_{ij,d}[\xi^0\mu^0_{j,d} + m_{ij,d}]}{2\sigma_{ij,d}} \\
&\propto -\frac{(\mu_{ij,d} - \tilde{\mu}_{ij})^2}{2\eta_{ij}\tilde{\sigma}_{ij,d}} \\
&= \mathcal{N}(\mu_{ij,d}|\tilde{\mu}_{ij,d}, \tilde{\xi_{ij}}\sigma_{ij,d}),
\end{aligned} \tag{A.31}$$

where we have defined

$$\tilde{\xi}_{ij} = \xi^0 + n_{ij}, \tag{A.32}$$

$$\tilde{\mu}_{ij,d} = \tilde{\xi}_{ij}^{-1}\left(\xi^0\mu^0_{ij,d} + m_{ij,d}\right). \tag{A.33}$$

We substitute for $\log q(\mu_{ij,d}, \sigma_{ij,d})$ using Eq. A.30, and we substitute for $\log q(\mu_{ij,d}|\sigma_{ij,d})$ using the result Eq, A.31. Keeping only the terms which

depend on $\sigma_{ij,d}$, we have

$$\begin{aligned}
&\log q(\sigma_{ij,d}) \\
&= -\frac{\xi^0(\mu_{ij,d}-\mu^0_{j,d})^2}{\sigma_{ij,d}} - \frac{1}{2}\log\sigma_{ij,d} + \frac{1}{2}\sigma^0_{j,d}{\sigma_{ij,d}}^{-1} + \left(\sum_{u,t}\gamma_{z_u=i}\gamma_{v_{ut}=j|z_u=i}\right)\sigma_{ij,d} \\
&\quad + \eta^0\sigma^0_{j,d} + \sum_{u,t}\gamma_{z_u=i}\gamma_{v_{ut}=j|z_u=i}\frac{(o_{ut}-\mu_{ij,d})^2}{\sigma_{ij,d}} + \frac{\tilde{\eta}_{ij}(o_{ut}-\tilde{\mu}_{ij,d})^2}{\tilde{\sigma}_{ij,d}} + \frac{1}{2}\tilde{\sigma}_{ij,d} \qquad \text{(A.34)} \\
&= \frac{1}{2}\tilde{\eta}_{ij}\sigma_{ij,d} - \frac{1}{2}\sigma_{ij,d}\tilde{\sigma}_{ij,d} \\
&\propto \mathcal{G}(\sigma_{ij,d}|\tilde{\eta}_{ij},\tilde{\sigma}_{ij,d}),
\end{aligned}$$

where we have defined

$$\tilde{\eta}_{ij} = \eta^0 + n_{ij}, \qquad \text{(A.35)}$$

$$\tilde{\sigma}_{ij,d} = \sigma^0_{j,d} + r_{ij,d} + \xi^0(\mu^0_{j,d})^2 - \tilde{\xi}_{ij}(\tilde{\mu}_{ij,d})^2. \qquad \text{(A.36)}$$

Using Eqs. A.31 and A.34 , Eq. 4.24 in section 4.3.1 is derived as follows:

$$\begin{aligned}
&\langle\frac{(o_{ut,d}-\mu_{ij,d})^2}{\sigma_{ij,d}}\rangle_{q(\mu_{ij,d}|\sigma_{ij,d})q(\sigma_{ij,d})} \\
&\quad = \int\int\frac{(o_{ut,d}-\mu_{ij,d})^2}{\sigma_{ij,d}}\mathrm{d}q\mathcal{N}(\mu_{ij}|\tilde{\mu}_{ij,d},\tilde{\xi}_{ij})\mathrm{d}q\mathcal{G}(\sigma_{ij}|\tilde{\eta}_{ij},\tilde{\sigma}_{ij}).
\end{aligned} \qquad \text{(A.37)}$$

Using the standard expressions for expectations under a Gaussian distribution, giving $\langle\mu_{ij,d}\rangle_{q(\mu_{ij,d})} = \tilde{\mu}_{ij,d}$ and $\langle\mu^2_{ij,d}\rangle_{q(\mu_{ij,d})} = \tilde{\mu}^2_{ij,d} + \tilde{\xi}_{ij}\tilde{\sigma}^{-1}_{ij,d}$, we obtain

$$\begin{aligned}
\left\langle(o_{ut,d}-\tilde{\mu}_{ij,d})^2\right\rangle_{q(\mu_{ij,d}|\sigma_{ij,d})} &= o^2_{ut,d} + \tilde{\xi}_{ij}\tilde{\sigma}^{-1}_{ij,d} - 2o_{ut,d}\tilde{\mu}_{ij,d} + \tilde{\mu}^2_{ij,d} \\
&= (o^2_{ut,d}-\tilde{\mu}_{ij,d})^2 + \tilde{\xi}_{ij}\tilde{\sigma}^{-1}_{ij,d}. \qquad \text{(A.38)}
\end{aligned}$$

Finally taking the expectation with respect to σ_{ij}, we obtain

$$\begin{aligned}
\left\langle\frac{(o_{ut,d}-\tilde{\mu}_{ij,d})^2}{\sigma_{ij,d}}\right\rangle_{q(\mu_{ij,d}|\sigma_{ij,d})q(\sigma_{ij,d})} &= \left\langle(o^2_{ut,d}-\tilde{\mu}_{ij,d})^2 + \tilde{\xi}_{ij}\tilde{\sigma}^{-1}_{ij,d}\right\rangle_{q(\sigma_{ij,d})} \\
&= \frac{\tilde{\eta}_{ij}(o_{ut,d}-\tilde{\mu}_{ij,d})^2 + \tilde{\xi}_{ij}}{\tilde{\sigma}_{ij,d}}. \qquad \text{(A.39)}
\end{aligned}$$

A.3 Measurements of speaker clustering evaluation

In this thesis, all of the speaker clustering experiments are evaluated with average cluster purity (ACP), the average speaker purity (ASP), and

their geometric mean value (K). Here, we give a brief explanation of the measurement [8]. The correct speaker labels for utterances were manually annotated. Let S_T be the correct number of speakers, S the estimated number of speakers, n_{ij} the estimated number of utterances assigned to speaker cluster i in all utterances of speaker j, n_j the estimated number of utterances of speaker j, n_i the estimated number of utterances assigned to speaker cluster i, and U the total number of utterances. Cluster purity p_i, speaker purity q_j, and the K value are then calculated as follows:

$$\begin{aligned} p_i &= \sum_{j=0}^{S_T} \frac{n_{ij}^2}{n_i^2}, \quad q_j = \sum_{i=0}^{S} \frac{n_{ij}^2}{n_j^2} m \\ K &= \sqrt{\frac{\sum_i p_i \cdot \sum_j q_j}{S_T S}}. \end{aligned} \tag{A.40}$$

We additionally calculated the speaker diarization error rate (DER) [73] in the experiments for CSJ. The DER is the ratio of incorrectly attributed speech time, which is calculated as follows

$$DER = \frac{U_{\text{fa}} + U_{\text{error}}}{U_{\text{ref}}}, \tag{A.41}$$

where U_{fa} denotes the total length of utterances not aligned with the speaker labels in the case where $S_T > S$ (i.e. false alarm utterances), namely the speech time of utterances assigned to improper speakers in the case that the estimated number of speakers is larger than the true number of speakers. U_{error} denotes the total length of utterances aligned with the wrong speaker labels and U_{ref} denotes the total length of all utterances in a test set. The clustering result and speaker labels concurred in order to minimize DER.

Bibliography

[1] J. Ajmera and C. Wooters. A robust speaker clustering algorithm. In *IEEE Automatic Speech Recognition Understanding Workshop*, 2003.

[2] L. Maaten and G. E. Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[3] M. A. Siegler, U. Jain, B. Raj, and R. M. Stern. Automatic segmentation, classification and clustering of broadcast news audio. In *Proc. DARPA Speech Recognition Workshop*, pages 97–99, Feb. 1997.

[4] S. S. Chen and P. S. Gopalakrishnan. Clustering via the bayesian information criterion with applications in speech recognition. In *ICASSP*, volume 2, pages 645–648, 1998.

[5] I. Lapidot. Som as likelihood estimator for speaker clustering. In *in Proc. Eurospeech*, 2003.

[6] A. Vandecatseye and J.-P. Martens. A fast, accurate and stream-based speaker segmentation and clustering algorithm. In *Proceedings of the 8th European Conference on Speech Communication and Technology*, number 14 in Topics in Speech Recognition and Segmentation, pages 941–944, Geneve, 9 2003. Causal Productions Pty Ltd.

[7] D. A. Reynolds, E. Singer, B. A. Carlson, G. C. O'Leary, J. J McLaughlin, and M. A. Zissman. Blind clustering of speech utterances based on speaker and language characteristics. In *ICSLP*. ISCA, 1998.

[8] A. Solomonoff, A. Mielke, M. Schmidt, and H. Gish. Clustering speakers by their voices. In *ICASSP*, pages 757–760, 1998.

[9] J. E. Rougui, M. Rziza, D. Aboutajdine, M. Gelgon, and J. M. Martínez. Fast incremental clustering of gaussian mixture speaker models for scaling up retrieval in on-line broadcast. In *ICASSP (5)*, pages 521–524. IEEE, 2006.

[10] M. Ben, M. Betser, F. Bimbot, and G. Gravier. Speaker diarization using bottom-up clustering based on a parameter-derived distance

between adapted gmms. In *in Intl. Conf. on Speech and Language Processing*, 2004.

[11] D. Moraru, M. Ben, and G. Gravier. Experiments on speaker tracking and segmentation in radio broadcast news. In *in European Conference on Speech Communication and Technology*, 2005.

[12] N. Tawara, S. Watanabe, Ogawa, T., and T. Kobayashi. Speaker clustering based on utterance-oriented Dirichlet process mixture model. In *INTERSPEECH*, pages 2905–2908, 2011.

[13] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *JOURNAL OF THE ROYAL STATISTICAL SOCIETY, SERIES B*, 39(1):1–38, 1977.

[14] Y. W. Teh, D. Newman, and M. Welling. A collapsed variational bayesian inference algorithm for latent dirichlet allocation. In P. B. Schölkopf, J. C. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 1353–1360. MIT Press, 2007.

[15] J. Sung, Z. Ghahramani, and S.-Y. Bang. Latent-space variational bayes. *IEEE Tran. on PAMI*, 30(12):2236–2242, 2008.

[16] J. Ø. Olsen. Separation of speakers in audio data. In *EUROSPEECH*, volume 1, 1995.

[17] E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S Willsky. A sticky HDP-HMM with application to speaker diarization. *Annals of Applied Statistics*, 5(2A):1020–1056, 2011.

[18] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag New York, Inc., 2006.

[19] G. McLachlan and D. Peel. *Finite Mixture Models*. John Wiley & Sons, 2004.

[20] J. L. Andrew, P. D. McNicholas, and S. Sudebi. Model-based classification via mixtures of multivariate t-distributions. *Computational Statistics and Data Analysis*, 55(1):520–529, 2011.

[21] A. Banerjee, I. S. Dhillon, J. Ghosh, and S. Sra. Clustering on the unit hypersphere using von Mises-Fisher distributions. *J. Mach. Learn. Res.*, 6:1345–1382, 2005.

[22] H. Tang, S. M. Chu, and T. S. Huang. Generative model-based speaker clustering via mixture of von Mises-Fisher distributions. In *ICASSP*, pages 4101–4104, 2009.

[23] J. S. Marron and M. P. Wand. Exact mean integrated squared error. *Ann. Stat.*, 20(2):712–736, 1992.

[24] C. J. Lawrence and W. J. Krzanowski. Mixture separation for mixed-mode data. *Statistics and Computing*, 6:85–92, March 1996.

[25] A. Willse and R. J. Boik. Identifiable finite mixtures of location models for clustering mixed-mode data. *Statist. & Computing 9,*, 9:111–121, 1999.

[26] D. G. Calo, A. Montanari, and C. Viroli. A hierarchical modeling approach for clustering probability density functions. *Computational Statistics and Data Analysis*, page 79–91, 2014.

[27] J. K. Vermunt and J. Magidson. Hierarchical mixture models for nested data structures. In *Classification: The Ubiquitous Challenge, in*. press. Heidelberg: Springer, 2005.

[28] J. K. Vermunt. A hierarchical mixture model for clustering three-way data sets. *Computational Statistics and Data Analysis*, 51(11):5368–5376, 2007.

[29] T. R. Belin. and D. B. Rubin. The analysis of repeated-measures data on schizophrenic reaction times using mixture models. *Statistics in Madicine*, 14:747–768, 1995.

[30] T. Hastie and R. Tibshirani. Discriminant analysis by gaussian mixtures. *Journal of the Royal Statistical Society, Series B*, 58:155–176, 1996.

[31] S. Watanabe, D. Mochihashi, T. Hori, and A. Nakamura. Gibbs sampling based multi-scale mixture model for speaker clustering. In *ICASSP*, pages 4524–4527, 2011.

[32] N. Tawara, T. Ogawa, S. Watanabe, and T. Kobayashi. Fully Bayesian inference of multi-mixture Gaussian model and its evaluation using speaker clustering. In *ICASSP*, pages 5253–5256, 2012.

[33] T. S. Ferguson. A bayesian analysis of some nonparametric problems. *Ann. Statist.*, 1(2):209–230, 1973.

[34] J. S. Liu. *Monte Carlo strategies in scientific computing*. Springer, jan 2008.

[35] D. Aldous. Exchangeability and related topics. E*cole d*et*e de probabilit*e*s de Saint-Flour*, XIII–1983:1–198, 1985.

[36] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren. DARPA TIMIT acoustic phonetic continuous speech corpus CDROM, 1993.

[37] T. Kawahara, H. Nanjo, and S. Furui. Automatic transcription of spontaneous lecture speech. In *Proc. IEEE workshop on Automatic Speech Recognition and Understanding*, 2001.

[38] L. Rabiner and B. H. Juang. *Fundamentals of Speech Recognition*. Signal processing. Prentice Hall, 1993.

[39] E. Spellman, B. C. Vemuri, and M. Rao. Using the KL-center for efficient and accurate retrieval of distributions arising from texture images. In *CVPR (1)*, pages 111–116, 2005.

[40] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker verification using adapted gaussian mixture models. In *Digital Signal Processing*, volume 10, pages 19–41, Jan. 2000.

[41] J. Diebolt and C. P. Robert. Estimation of finite mixture distributions through Bayesian sampling. *J. R. Statist. Soc B*, 56(2):363–375, 1994.

[42] C. P. Robert and G Casella. *Monte Carlo Statistical Methods, 2nd ed.* Springer, 2004.

[43] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, March 2003.

[44] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[45] F. Valente and C. J Wellekens. Variational bayesian adaptation for speaker clustering. In *ICASSP*, 03 2005.

[46] S. Itahashi. On recent speech corpora activities in japan. *Journal of the Acoustical Society of Japan (E)*, 20(3):163–169, 1999.

[47] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet. Front-end factor analysis for speaker verification. *IEEE Transaction Speech Audio Process.*, 19(4):788–798, 2011.

[48] S. Shum, N. Dehak, E. Chuangsuwanich, D. Reynolds, and J. Glass. Exploiting intra-conversation variability for speaker diarization. In *interspeech*, pages 945–948, Aug. 2011.

[49] S. Shum, N. Dehak, E. Chuangsuwanich, D. Reynolds, and J. Glass. On the use of spectral and iterative methods for speaker diarization. In *Interspeech*, Sept. 2012.

[50] H. Ning, M. Liu, H. Tang, and T. Huang. A spectral clustering approach to speaker diarization. In *ICSLP*, May 2006.

[51] K. Iso. Speaker clustering using vector quantization and spectral clustering. In *ICASSP*, pages 4986–4989, Mar. 2010.

[52] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS*, Dec. 2001.

[53] S. Itahashi. A noise database and japanese common speech data corpus. *Journal of the Acoustical Society of Japan*, 47(12):951–953, 1991. in Japanese.

[54] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Popular, classical, and jazz music database. In *3rd International Conference on Music Information Retrieval (ISMIR)*, pages 287—288, Oct. 2002.

[55] K. Itou, M. Yamamoto, K. Takeda, T. Takezawa, T. Matsuoka, T. Kobayashi, K. Shikano, and S. Itahashi. JNAS: Japanese speech corpus for large vocabulary continuous speech recognition research. *Acoust Soc Jpn E*, 20(3):199–206, 1999.

[56] ASJ continuous speech corpus for research (ASJ-JIPDEC). *National Institute of Information*. http://research.nii.ac.jp/src/en/ASJ-JIPDEC.html.

[57] A. E. Gelfand A. Rodriguez, D. B. Dunson. The nested Dirichlet process. *Journal of the American Statistical Association*, 103:1131–1154, September 2008.

[58] N. Tawara, T. Ogawa, S. Watanabe, A. Nakamura, and T. Kobayashi. Fully Bayesian speaker clustering based on hierarchically structured utterance-oriented Dirichlet process mixture model. In *INTERSPEECH*, 2012.

[59] N. Tawara, T. Ogawa, S. Watanabe, A. Nakamura, and T. Kobayashi. A sampling-based speaker clustering using utterance-oriented dirichlet process mixture model and its evaluation on large scale data. *APSIPA Transactions on Signal and Information Processing*, 2015.

[60] A. Y. Ng, M. I. Jordan, and Y. Weiss. Self-tuning spectral clustering. In *NIPS*, pages 1601—1608, Dec. 2004.

[61] R. Mall, H. Bensmail, R. Langone, C. Varon, and J. A. K. Suykens. Denoised kernel spectral data clustering. In *2016 International Joint Conference on Neural Networks, IJCNN 2016*, pages 3709–3716, 2016.

[62] T. Xiang and S. Gong. Spectral clustering with eigenvector selection. *Pattern Recogn.*, 41(3):1012–1029, March 2008.

[63] F. Zhao, L. Jiao, H. Liu, X. Gao, and M. Gong. Spectral clustering with eigenvector selection based on entropy ranking. *Neurocomput.*, 73(10-12):1704–1717, June 2010.

[64] N. Rebagliati and A. Verri. Spectral clustering with more than k eigenvectors. *Neurocomput.*, 74(9):1391–1401, April 2011.

[65] V. Hautamaki P. Rajan, T. Kinnunen. Effect of multicondition training on i-vector PLDA configurations for speaker recognition. In *Interspeech*, pages 3694–3697, 2013.

[66] P. Kenny. Bayesian speaker verification with heavy- tailed priors. In *Odyssey: The Speaker and Language Recognition Workshop*, June 2010.

[67] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, O. Friedland, and O. Vinyals. Speaker diarization : A review of recent research. *"IEEE Trans. On Audio, Speech, and Language Processing" (TASLP), special issue on "New Frontiers in Rich Transcription", February 2012, Volume 20, N°2, ISSN: 1558-7916*, 05 2011.

[68] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 5 2015.

[69] G. Heigold, P. Nguyen, M. Weintraub, and V. Vanhoucke. Investigations on exemplar-based features for speech recognition towards thousands of hours of unsupervised, noisy data. In *ICASSP*, pages 4437–4440. IEEE, 2012.

[70] Y. Zhang and J. R. Glass. Unsupervised spoken keyword spotting via segmental dtw on gaussian posteriorgrams. In *ASRU*, pages 398–403. IEEE, 2009.

[71] Y. Zhang, R. Salakhutdinov, H. A. Chang, and J. R. Glass. Resource configurable spoken query detection using deep boltzmann machines. In *ICASSP*, pages 5161–5164. IEEE, 2012.

[72] A. Jansen, E. Dupoux, S. Goldwater, M. Johnson, S. Khudanpur, K. Church, N. Feldman, H. Hermansky, F. Metze, R. C. Rose, M. Seltzer, P. Clark, I. McGraw, B. Varadarajan, E. Bennett, B. Börschinger, J. Chiu, E. Dunbar, A. Fourtassi, D. Harwath, C. ying Lee, K. Levin, A. Norouzian, V. Peddinti, R. Richardson, T. Schatz, and S. Thomas. A summary of the 2012 jhu clsp workshop on zero resource speech technologies and models of early language acquisition. In *ICASSP*, pages 8111–8115. IEEE, 2013.

[73] J. G. Fiscus, J. Ajot, and J. S. Garofolo. The rich transcription 2007 meeting recognition evaluation. In *CLEAR*, pages 373–389, 2007.

List of works

Journal papers

- Naohiro Tawara, Tetsuji Ogawa, Shinji Watanabe, Tetsunori Kobayashi, “Nested Gibbs sampling for mixture-of-mixture model and its application to speaker clustering,” APSIPA Transactions on Signal and Information Processing, vol.5:e16, pp. 1-13, Aug. 2016
- Naohiro Tawara, Tetsuji Ogawa, Shinji Watanabe, Atsushi Nakamura, Tetsunori Kobayashi, “A sampling-based speaker clustering using utterance-oriented Dirichlet process mixture model and its evaluation on large scale data”, APSIPA Transactions on Signal and Information Processing, vol.4:e6, pp. 1-10, Sept. 2015

International conferences

- Naohiro Tawara, Tetsuji Ogawa, Tetsunori Kobayashi, “A comparative study of spectral clustering for i-vector-based speaker clustering under noisy conditions,” Proceedings of ICASSP2015, MLSP-P2.9, March 2015.
- Naohiro Tawara, Tetsuji Ogawa, Shinji Watanabe, Atsushi Nakamura, Tetsunori Kobayashi, “Gibbs sampling based multi-scale mixture model for speaker clustering on noisy data,” Proceedings of MLSP2013, Sept. 2013.
- Naohiro Tawara, Tetsuji Ogawa, Shinji Watanabe, Atsushi Nakamura, Tetsunori Kobayashi, “Fully Bayesian speaker clustering based on hierarchically structured utterance-oriented Dirichlet process mixture model,” Proceedings of Interspeech2012, Thu.O9b.04, Sept. 2012.
- Naohiro Tawara, Tetsuji Ogawa, Shinji Watanabe, Atsushi Nakamura, Tetsunori Kobayashi, “Fully Bayesian inference of multi-mixture Gaussian model and its evaluation using Speaker clustering,” Proceedings of ICASSP2012, pp. 5253-5256, March 2012.

- Naohiro Tawara, Shinji Watanabe, Tetsuji Ogawa, Tetsunori Kobayashi, “Speaker clustering based on utterance-oriented Dirichlet process mixture model,” Proceedings of Interspeech2011, pp.2905-2908, Aug. 2011.

Domestic conferences

- Naohiro Tawara, Tetsuji Ogawa, Tetsunori Kobayashi, ”Studying effect of factor analysis on spectral-clustering based speaker clustering,” Proceedings of the 2015 Autumn Meeting of the Acoustical Society of Japan, pp. 173-174, Sept. 2015 (in Japanese).
- 俵直弘, 小川哲司, 小林哲則, “i-vectorを用いたスペクトラルクラスタリングによる雑音環境下話者クラスタリング,” The Special Interest Group Technical Reports of IPSJ, 2015-SLP-105, no.11, pp.1-6, Feb. 2015 (in Japanese).
- Naohiro Tawara, Tetsuji Ogawa, Tetsunori Kobayashi, ”Speaker clustering based on spectral clustering under noisy circumstances,” Proceedings of the 2014 Autumn Meeting of the Acoustical Society of Japan, pp. 95-98, Sept. 2014 (in Japanese).
- Yusuke, Fukuchi, Naohiro Tawara, Tetsuji Ogawa and Tetsunori Kobayashi, ”Survey of Speaker Verification with Factor Analysis in Robustness for Environment,” Proceedings of the 2013 Autumn Meeting of the Acoustical Society of Japan, pp. 75-78, Sept. 2013 (in Japanese).
- 俵直弘 , 小川哲司 , 渡部晋治 , 中村篤, 小林哲則, “効率的なサンプリング手法を用いた話者モデリング,” The Special Interest Group Technical Reports of IPSJ, Vol. 2013-SLP-97, No.2, July 2013 (in Japanese).
- Yusuke Fukuchi, Naohiro Tawara, Tetsuji Ogawa and Tetsunori Kobayashi, ”Speaker clustering based on non-negative matrix factorization using i-vector-based speaker similarity,” The Special Interest Group Technical Reports of IPSJ, Vol.2012-SLP-92, No.8, July 2012 (in Japanese).
- Naohiro Tawara, Tetsuji Ogawa, Shinji Watanabe, Atsushi Nakamura and Tetsunori Kobayashi, ”Fully Bayesian speaker clustering based on hierarchical structured Dirichlet process mixture model,” IEICE technical report, vol.110, no.76, pp. 21-28, March 2012 (in Japanese).

- Naohiro Tawara, Tetsuji Ogawa, Shinji Watanabe, Atsushi Nakamura and Tetsunori Kobayashi, ”Fully Bayesian speaker clustering with utterance oriented DPMM and its evaluation on large scale data,” Proceedings of the 2012 Spring Meeting of the Acoustical Society of Japan, pp. 207-210, March 2012 (in Japanese).
- 俵直弘, 小川哲司, 渡部晋治, 小林哲則, “階層的発話生成モデルを用いた話者クラスタリングのためのフルベイズモデル推定手法の比較,” The 14th Information-Based Induction Sciences Workshop (IBIS2011), March 2012 (in Japanese).
- Naohiro Tawara, Shinji Watanabe, Tetsuji Ogawa and Tetsunori Kobayashi, ”Fully Bayesian inference of multi-mixture Gaussian model and its evaluation using speaker clustering,” Proceedings of the 2011 Autumn Meeting of the Acoustical Society of Japan, pp. 175-178, Sept. 2011 (in Japanese).
- Naohiro Tawara, Shinji Watanabe, Tetsuji Ogawa and Tetsunori Kobayashi, ”Speaker clustering based on utterance-oriented Dirichlet process mixture model,” Proceedings of the 2011 Spring Meeting of the Acoustical Society of Japan, pp. 41-44, March 2011 (in Japanese).

Others

- Won a student award in 2011 Spring Meeting of the Acoustical Society of Japan, March 2011